



Программный комплекс Систэм Платформ

---

## SePlatform.Data Server 2.1 Модуль Modbus RTU Master

---

Руководство администратора

Редакция  
3. Предварительная

Соответствует версии ПО  
2.1.2

---



© ООО «СИСТЭМ СОФТ», 2022-2024. Все права защищены.

Авторские права на данный документ принадлежат ООО «СИСТЭМ СОФТ». Копирование, перепечатка и публикация любой части или всего документа не допускается без письменного разрешения правообладателя.

# Содержание

<b>1. Назначение и принцип работы</b>	<b>4</b>
1.1. Обмен данными по протоколу Modbus	4
1.1.1. Типы данных сигналов SePlatform.Data Server	5
1.1.2. Протокольные типы	5
1.1.3. Сегменты памяти подчиненной станции	7
1.1.4. Поддерживаемые функции протокола Modbus	8
1.1.4.1. Чтение данных	8
1.1.4.2. Запись одного значения	11
1.1.4.3. Запись нескольких значений	13
1.2. Опрос подчиненной станции	15
1.2.1. Качество сигналов	16
1.2.2. Метка времени	17
1.3. Стратегия формирования запросов на чтение	18
1.4. Отправка управляющих и регулирующих воздействий	19
1.4.1. Сигнал доставки команды	19
1.5. Работа модуля в резерве	20
<b>2. Настройка обмена данными со станцией</b>	<b>21</b>
2.1. Исходные данные для настройки	21
2.2. Пример для настройки обмена данными	21
2.3. Настройка в SePlatform.Development Studio	22
2.3.1. Настройка станции Modbus (ПЛК)	22
2.3.1.1. Настройка нескольких станций	26
2.3.2. Настройка Modbus RTU Master	26
2.3.2.1. Настройка взаимодействия со станцией Modbus	28
2.3.2.2. Настройка категорий данных	31
2.3.3. Настройка канала для обмена данными	32
2.3.3.1. Последовательный интерфейс RS-232/422/485	32
2.3.3.2. Шлюз Ethernet/COM (Modbus RTU over TCP)	34
2.3.4. Настройка обмена данными	35
2.3.4.1. Настройка логических типов	36
2.3.4.2. Получение и отправка значений сигналов	39
2.3.4.3. Настройка сигнала доставки	40
2.3.5. Применение конфигурации SePlatform.Data Server	40
<b>3. Диагностика работы модуля</b>	<b>42</b>
3.1. Служебные сигналы	42
3.2. Статистика	42
3.3. Журнал работы модуля	44
3.3.1. Структура кадров модуля	45
3.3.1.1. Кадры функций чтения	46
3.3.1.2. Кадры функций записи	50
3.3.1.3. Кадры ошибок	53
<b>Список терминов и сокращений</b>	<b>55</b>

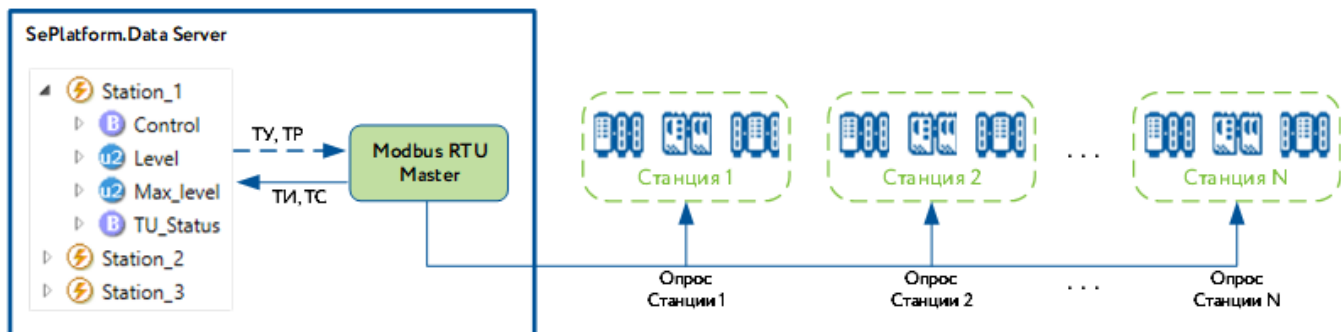
# 1. Назначение и принцип работы

Модуль Modbus RTU Master - коммуникационный модуль, предназначенный для обмена данными между SePlatform.Data Server и подчиненными станциями по протоколу Modbus RTU.

Функции модуля Modbus RTU Master:

- сбор данных: модуль записывает в сигналы SePlatform.Data Server значения, полученные от подчиненной станции;
- подача команд управления: модуль передает значения сигналов SePlatform.Data Server в подчиненную станцию.

Модуль Modbus RTU Master обменивается данными с подчиненными станциями через последовательный интерфейс RS-232/422/485 (COM-порт) или шлюз Ethernet/COM (Modbus RTU over TCP). Опрос станций выполняется последовательно.



## 1.1. Обмен данными по протоколу Modbus

Обмен данными по протоколу Modbus основан на архитектуре Master - Slave. Модуль Modbus RTU Master выполняет обмен данными с подчиненными станциями, реализуя функции Master в соответствии со спецификацией MODBUS APPLICATION PROTOCOL SPECIFICATION.

Модуль инициирует обмен данными, отправляя запрос подчиненной станции. Подчиненная станция отвечает на запрос, предоставляя запрошенные данные или выполняя команду модуля. Самостоятельно подчиненная станция данные не передает.

Запросы и ответы представляют собой кадры данных. Для модуля Modbus RTU Master запрос, отправляемый подчиненной станции, является исходящим кадром, а ответ, полученный от подчиненной станции - входящим кадром.

Исходящий кадр содержит код функции и данные, необходимые для выполнения функции. Входящий кадр содержит код функции и данные, полученные в результате выполнения функции. Подробное описание кадров данных модуля Modbus RTU Master приведено в разделе Журнал работы модуля ([стр. 44](#)).













Для обмена данными с подчиненной станцией модуль Modbus RTU Master, в зависимости от типа данных сигнала и направления передачи данных, преобразует тип сигнала к протокольному типу, а в зависимости от сегмента памяти, указанного в адресе сигнала, определяет функцию Modbus и формирует кадр данных.

### 1.1.1. Типы данных сигналов SePlatform.Data Server

Тип данных определяет множество значений, которые может принимать сигнал.

Для обмена данными с подчиненной станцией тип данных сигнала SePlatform.Data Server должен совпадать с типом данных соответствующего сигнала подчиненной станции.

Типы данных сигналов SePlatform.Data Server:

Тип	Описание	Допустимые значения
 Int1	Знаковое целое 1 байт	[-128; 127]
 UInt1	Беззнаковое целое 1 байт	[0; 255]
 Int2	Знаковое целое 2 байта	[-32 768; 32 767]
 UInt2	Беззнаковое целое 2 байта	[0; 65 535]
 Int4	Знаковое целое 4 байта	[-2 147 483 648; 2 147 483 647]
 UInt4	Беззнаковое целое 4 байта	[0; 4 294 967 295]
 Int8	Знаковое целое 8 байт	[-9 223 372 036 854 775 808; 9 223 372 036 854 775 807]
 UInt8	Беззнаковое целое 8 байт	[0; 18 446 744 073 709 551 615]
 Float	Значение с плавающей запятой 4 байта	$[\pm 1.5 \times 10^{-45}; \pm 3.4 \times 10^{38}]$ . Точность 6-9 цифр
 Double	Значение с плавающей запятой 8 байт	$[\pm 5.0 \times 10^{-324}; \pm 1.7 \times 10^{308}]$ . Точность 15-17 цифр
 Bool	Логическое значение	true, false
 String	Текстовая строка в кодировке UTF16	до 2 миллиардов знаков, каждый знак занимает 16 бит (2 байта)

### 1.1.2. Протокольные типы

Протокольный тип определяет направление передачи и размер передаваемых данных.













В таблице приведены протокольные типы, используемые модулем Modbus RTU Master для передачи данных в направлениях:













- Master → Slave: данные передаются от модуля к подчиненной станции (управляющие команды);
- Slave → Master: данные передаются от подчиненной станции к модулю (чтение данных).

Протокольный тип	Тип значения	Описание
Master → Slave		

Протокольный тип	Тип значения	Описание
TC	bool	Телеуправление
TR2	int2/uint2	Телерегулирование
TR2_TIME	int2/uint2	Телерегулирование с меткой времени
TR4	int4/uint4	Телерегулирование
TR4_TIME	int4/uint4	Телерегулирование с меткой времени
TRF4	float	Телерегулирование
TRF4_TIME	float	Телерегулирование с меткой времени
STR-COMMAND	string	Телерегулирование
Slave → Master		
TS	bool	Телесигнализация
TM2	int2/uint2	Телеизмерение
TM2_TIME	int2/uint2	Телеизмерение с меткой времени
TMC	int4/uint4	Телесчет
TMC_TIME	int4/uint4	Телесчет с меткой времени
TMF4	float	Телеизмерение
TMF4_TIME	float	Телеизмерение с меткой времени
TMF8	double	Телеизмерение
TMF8_TIME	double	Телеизмерение с меткой времени
STR	string	Телеизмерение

Каждый тип сигнала SePlatform.Data Server преобразуется модулем Modbus RTU Master к нескольким протокольным типам в зависимости от указанного в адресе сигнала сегмента памяти. В таблице приведены типы сигналов SePlatform.Data Server и допустимые для них протокольные типы.

Протокольный тип	Тип в SePlatform.Data Server											
	 i1	 u1	 i2	 u2	 i4	 u4	 i8	 u8	 r4	 r8	 S	 B
	int1	uint1	int2	uint2	int4	uint4	int8	uint8	float	double	string	bool
TS												+
TC												+
TM2, TM2_TIME			+	+	+	+	+	+	+	+		

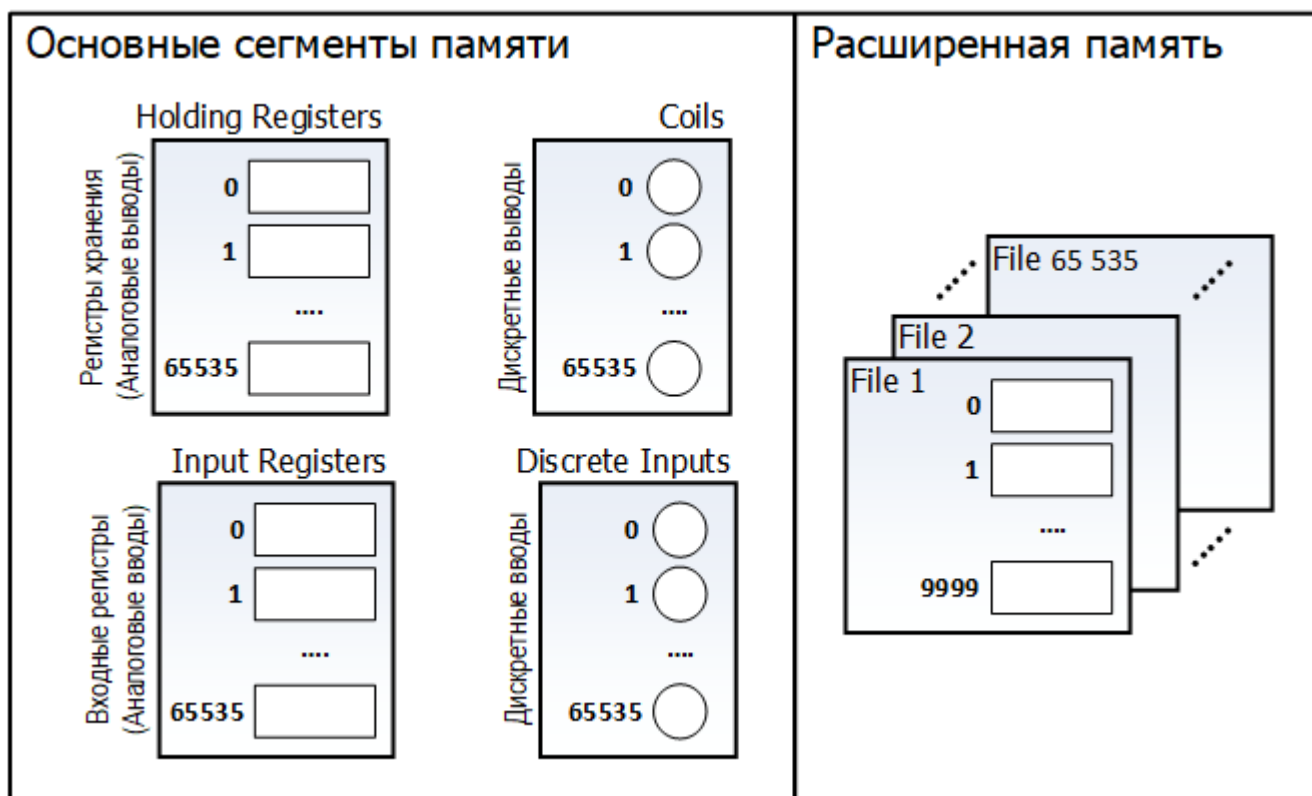
Протокольный тип	Тип в SePlatform.Data Server											
	 int1	 uint1	 int2	 uint2	 int4	 uint4	 int8	 uint8	 float	 double	 string	 bool
TMF4, TMF4_TIME									+	+		
TMC, TMC_TIME					+	+	+	+	+	+		
TMF8, TMF8_TIME										+		
TR2, TR2_TIME			+	+								
TRF4, TRF4_TIME			+	+	+	+			+			
TR4, TR4_TIME			+	+	+	+	+	+				
TRF8, TRF8_TIME										+		
STR											+	
STR-COMMAND											+	

### 1.1.3. Сегменты памяти подчиненной станции

Модуль Modbus RTU Master выполняет чтение и запись данных в четыре основных сегмента памяти, а также чтение данных из файлов расширенной памяти подчиненной станции. В таблице приведены сегменты памяти подчиненной станции, тип и размер элементов сегмента, а также действия, допустимые для сегментов.

Сегмент	Описание	Тип и размер элемента	Действия
Coils	Дискретные выходы (Телеуправление)	Ячейка 1 бит	Запись и чтение
Discrete Inputs	Дискретные входы (Телесигнализация)	Ячейка 1 бит	Чтение
Holding Registers	Регистры хранения (Телерегулирование)	Регистр 2 байта	Запись и чтение
Input Registers	Регистры ввода (Телеизмерения)	Регистр 2 байта	Чтение
Files	16-битные регистры, организованные в файлы, состоящие из записей произвольной длины	Регистр 2 байта	Чтение

Каждый сегмент памяти содержит «65 536» элементов, адреса которых от «0» до «65 535». Расширенная память содержит файлы с номерами от «1» до «65 535», каждый из которых содержит от «0» до «9999» записей.



### 1.1.4. Поддерживаемые функции протокола Modbus

Стандартные функции протокола Modbus, используемые модулем Modbus RTU Master для обмена данными с подчиненной станцией, приведены в таблице.

Код функции	Название	Действие	Сегмент
«01 (0x01)»	«Read Coils»	Чтение состояния дискретных выводов	Coils
«02 (0x02)»	«Read Discrete Inputs»	Чтение состояния дискретных вводов	Discrete Inputs
«03 (0x03)»	«Read Holding Registers»	Чтение регистров хранения	Holding Registers
«04 (0x04)»	«Read Input Registers»	Чтение входных регистров	Input Registers
«05 (0x05)»	«Write Single Coil»	Запись одного дискретного вывода	Coils
«06 (0x06)»	«Write Single Register»	Запись одного регистра хранения	Holding Registers
«16 (0x10)»	«Write Multiple registers»	Запись нескольких регистров хранения	Holding Registers
«20 (0x14)»	«Read File Record»	Чтение данных из файла	Files

#### 1.1.4.1. Чтение данных

Для чтения данных из сегментов памяти подчиненной станции используются стандартные функции протокола Modbus:

- «01 (0x01) Read Coils» - чтение состояния дискретных выводов;
- «02 (0x02) Read Discrete Inputs» - чтение состояния дискретных вводов;



- «03 (0x03) Read Holding Registers» - чтение регистров хранения;
- «04 (0x04) Read Input Registers» - чтение входных регистров;
- «20 (0x14) Read File Record» - чтение данных из файла.

Максимальное количество считываемых элементов за один запрос для каждого сегмента задаётся в параметрах модуля ([стр. 30](#)).

Структура кадров данных приведена в разделе Кадры функций чтения ([стр. 46](#)).

В таблице приведены типы сигналов SePlatform.Data Server, соответствующие им протокольные типы для чтения данных подчиненной станции и функции, выполняющие чтение данных из сегментов памяти станции.

Тип сигнала	Протокольные типы	Функция
Bool	TS	«01 (0x01) Read Coils» «02 (0x02) Read Discrete Inputs» «03 (0x03) Read Holding Registers» «04 (0x04) Read Input Registers»
Int2/UInt2	TM2, TM2_TIME	«03 (0x03) Read Holding Registers» «04 (0x04) Read Input Registers» «20 (0x14) Read File Record»
Int4/UInt4	TM2, TM2_TIME, TMC, TMC_TIME	
Int8/UInt8	TM2, TM2_TIME, TMC, TMC_TIME	
Float	TM2, TM2_TIME, TMF4, TMF4_TIME, TMC, TMC_TIME	
Double	TM2, TM2_TIME, TMF4, TMF4_TIME, TMC, TMC_TIME, TMF8, TMF8_TIME	
String	STR	

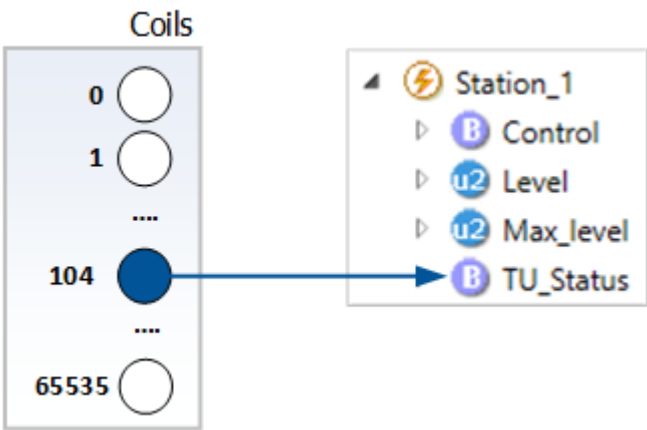
Чтобы получать данные от подчиненной станции:

- добавьте параметр с направлением «выход»;
- в Карте адресов Modbus для добавленного параметра укажите:
  - сегмент памяти, из которого считываются данные;
  - адрес элемента сегмента памяти (для сегмента Files - номер файла и номер записи).

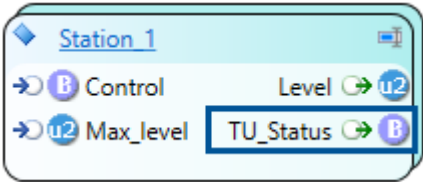


ПРИМЕР

От подчиненной станции требуется получать значение ячейки с адресом «104».



В логический тип «Station\_1» добавьте параметр «TU\_Status» типа Bool с направлением «выход»:



В Карте адресов Modbus для сигнала «TU\_Status» укажите:

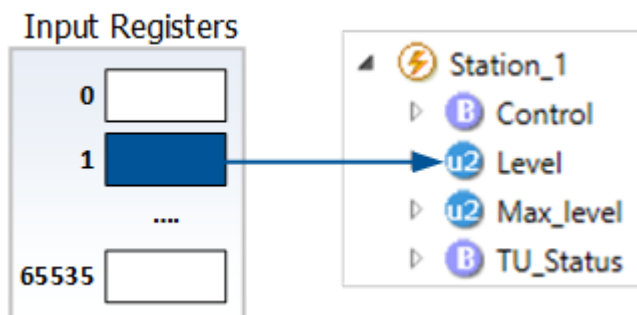
- > Сегмент - «Coils»;
- > Адрес - «104».

My-project.Domain.PLC_1.CPU.App_PLC_1.ModbusAddressMap Карта адресов Modbus										
	Сигнал	Тип	Привязка	Сегмент	Адрес	Номер бита	Номер записи в файле	Метка времени	Размер строки	Категория дан
u2	Station_1.Level	uint2	не привязан							
B	Station_1.TU_Status	bool	непосредственно	Coils	104					

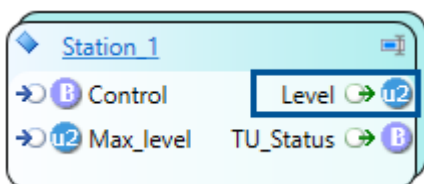


## ПРИМЕР

От подчиненной станции требуется получать значение регистра с адресом «1», диапазон значений которого от «0» до «4000».



В логический тип «Station\_1» добавьте параметр «Level» типа Uint2 с направлением «выход»:



В Карте адресов Modbus для сигнала «Level» укажите:

- Сегмент - «Input Registers»;
- Адрес - «1».

My-project.Domain.PLC 1.CPU.App PLC 1. ModbusAddressMap Карта адресов Modbus										
	Сигнал	Тип	Привязка	Сегмент	Адрес	Номер бита	Номер записи в файле	Метка времени	Размер строки	Категория дан
	Station_1.Level	uint2	непосредственно	Input Registers	1					
	Station_1.TU_Status	bool	непосредственно	Coils	104					

### 1.1.4.2. Запись одного значения

Для записи одного значения в сегмент памяти подчиненной станции используются стандартные функции протокола Modbus:

- «05 (0x05) Write Single Coil» - запись значения одного дискретного вывода;
- «06 (0x06) Write Single Register» - запись значения в один регистр хранения.



## ОБРАТИТЕ ВНИМАНИЕ

Чтобы использовать функцию «06 (0x06) Write Single Register» для записи значений в регистры хранения, в свойствах станции параметру **Запись в регистры** установите значение «Single».

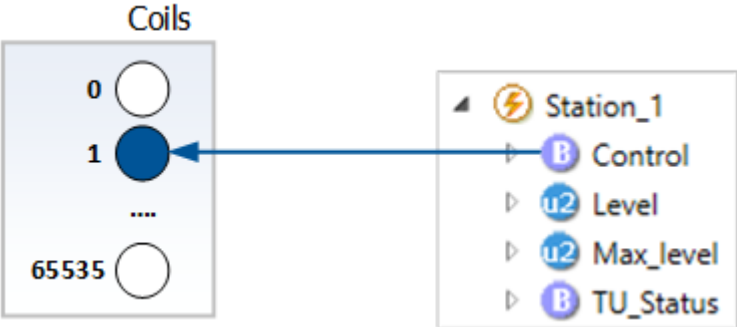
Структура кадров данных приведена в разделе Кадры функций записи ([стр. 50](#)).

В таблице приведены типы сигналов SePlatform.Data Server, соответствующие им протокольные типы для передачи команд подчиненной станции и функции, выполняющие запись данных в сегменты памяти станции.

Тип сигнала сервера	Протокольные типы	Функция
Bool	TC	«05 (0x05) Write Single Coil»
Int2/Uint2	TR2	«06 (0x06) Write Single Register»

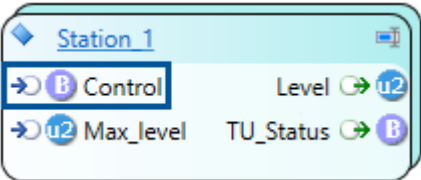
Чтобы записывать данные в подчиненную станцию:

- добавьте параметр с направлением «вход»;
- в Карте адресов Modbus для добавленного параметра укажите:
  - сегмент памяти, в который требуется записать данные;
  - адрес изменяемого элемента сегмента памяти.



В подчиненной станции требуется изменять значение ячейки с адресом «1».

В логический тип «Station\_1» добавьте параметр «Control1» типа Bool с направлением «вход»:



В Карте адресов Modbus для сигнала «Control1» укажите:

- Сегмент - «Coils»;
- Адрес - «1».

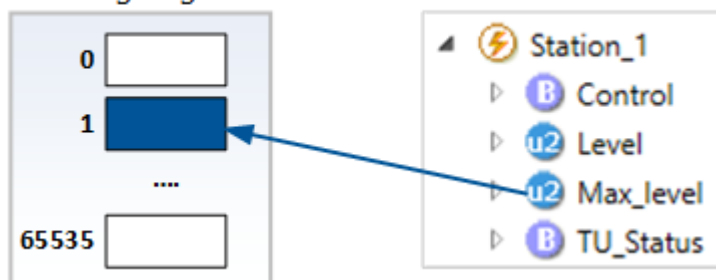
	Сигнал	Тип	Привязка	Сегмент	Адрес	Номер бита	Номер записи в файле	Метка времени	Размер строки	Категория дан
u2	Station_1.Level	uint2	непосредственно	Input Registers	1					
B	Station_1.TU_Status	bool	непосредственно	Coils	104					
B	Station_1.Control	bool	непосредственно	Coils	1					
u2	Station_1.Max_level	uint2	не привязан							



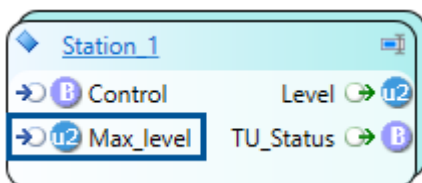
## ПРИМЕР

В подчиненной станции требуется изменять значение регистра с адресом «1», диапазон значений которого от «0» до «4000».

## Holding Registers



В логический тип «Station\_1» добавьте параметр «Max\_level» типа Uint2 с направлением «вход»:



В Карте адресов Modbus для сигнала «Max\_level» укажите:

- Сегмент - «Holding Registers»;
- Адрес - «1».

My-project.Domain.PLC 1.CPU.App PLC 1. ModbusAddressMap Карта адресов Modbus

	Сигнал	Тип	Привязка	Сегмент	Адрес	Номер бита	Номер записи в файле	Метка времени	Размер строки	Категория дан
u2	Station_1.Level	uint2	непосредственно	Input Registers	1					
B	Station_1.TU_Status	bool	непосредственно	Coils	104					
B	Station_1.Control	bool	непосредственно	Coils	1					
u2	Station_1.Max_level	uint2	непосредственно	Holding Registers	1					

### 1.1.4.3. Запись нескольких значений

Для записи значений в несколько регистров памяти подчиненной станции используется стандартная функция протокола Modbus «16 (0x10) Write Multiple Registers».



## ОБРАТИТЕ ВНИМАНИЕ

Чтобы использовать функцию «16 (0x10) Write Multiple Registers» для записи значений в регистры хранения, в свойствах станции параметру **Запись в регистры** установите значение «Multiple».

Максимальное количество записываемых элементов за одну команду задаётся в параметрах модуля.

Структура кадров данных приведена в разделе Кадры функций записи ([стр. 50](#)).

В таблице приведены типы сигналов SePlatform.Data Server и соответствующие им протокольные типы для передачи команд подчиненной станции с помощью функции «16 (0x10) Write Multiple Registers».

Тип сигнала	Протокольные типы
Int2/UInt2	TR2_TIME, TR4, TR4_TIME, TRF4, TRF4_TIME
Int4/UInt4	TR4, TR4_TIME, TRF4, TRF4_TIME
Int8/UInt8	TR4, TR4_TIME
Float	TRF4, TRF4_TIME
Double	TRF8, TRF8_TIME
String	STR-COMMAND

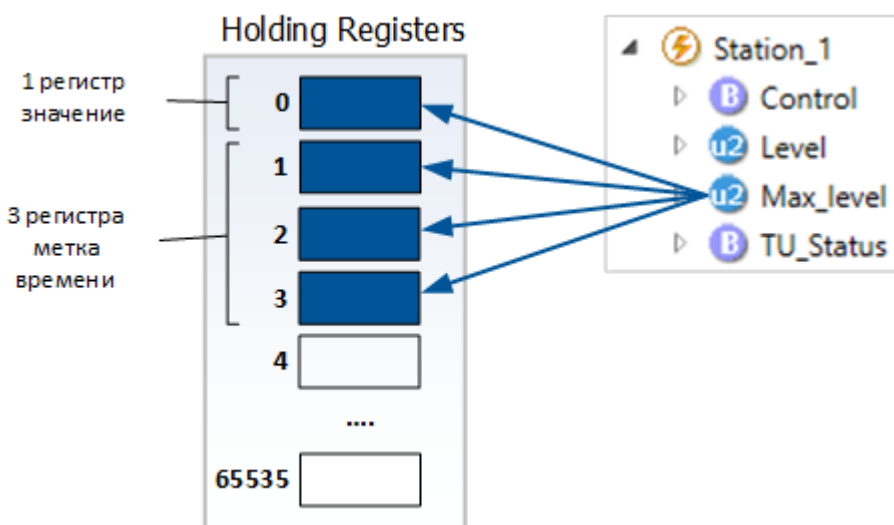
Чтобы отправлять данные в подчиненную станцию:

- добавьте параметр с направлением «**вход**»;
- в **Карте адресов Modbus** для добавленного параметра укажите:
  - сегмент памяти, в который требуется записать данные;
  - адрес изменяемого элемента сегмента памяти.

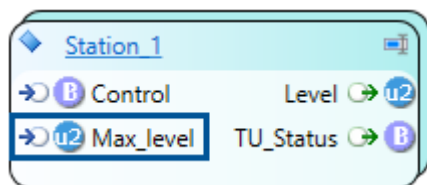


## ПРИМЕР

В подчиненной станции требуется изменять значения четырёх регистров начиная с адреса «0» (1 регистр - значение, 3 регистра - метка времени).



В логический тип «Station\_1» добавьте параметр «Max\_level» типа Uint2 с направлением «вход»:



В Карте адресов Modbus для сигнала «Max\_level» укажите:

- Сегмент - «Holding Registers»;
- Адрес - «0»;
- Метка времени - «Да».

	Сигнал	Тип	Привязка	Сегмент	Адрес	Номер бита	Номер записи в файле	Метка времени	Размер строки	Категория дан
u2	Station_1.Level	uint2	непосредственно	Input Registers	1					
B	Station_1.TU_Status	bool	непосредственно	Coils	104					
B	Station_1.Control	bool	непосредственно	Coils	1					
u2	Station_1.Max_level	uint2	непосредственно	Holding Registers	0			Да		

## 1.2. Опрос подчиненной станции

Опрос нескольких подчиненных станций ведется модулем Modbus RTU Master по очереди, порядок станций определяется таблицей поллинга.

Модуль отправляет запросы каждой подчиненной станции последовательно: отправляет запрос и ждет ответа. Время между отправкой запросов задается в параметрах модуля.

Когда запрос отправлен подчиненной станции, запускается счетчик **Времени ожидания ответа от станции**. При получении ответа от подчиненной станции модуль проверяет его на ошибки. В случае ошибки **Время ожидания ответа от станции** истекает и выставляется состояние потери связи со станцией, качество сигнала принимает значение **COMM\_FAILURE**.

Максимальное количество повторов запросов на чтение и максимальное количество повторов на отправку команд задается в конфигурации модуля. Когда количество повторов превышает заданное максимальное количество, то регистрируется потеря связи со станцией.

Если в ответ на запрос подчиненная станция вернула строку **SLAVE\_DEVICE\_BUSY**, то повтор отправки запроса будет произведен после **Паузы опроса занятого устройства**. При этом необходимо чтобы **Пауза опроса занятого устройства** была меньше, чем **Время ожидания ответа от станции**.

При опросе и выдаче управляющих воздействий модуль позволяет изменять порядок байт. Функция изменения порядка байт необходима для работы с устройствами, которые на аппаратном уровне меняют местами байты в словах ([стр. 24](#)).

### 1.2.1. Качество сигналов

В протоколе Modbus отсутствует понятие качества данных, поэтому качество устанавливается модулем исходя из результатов выполнения запросов. При старте модуля все сигналы имеют плохое качество.

Качества сигналов, выставляемых модулем, приведены в таблице:

Значение качества	Идентификатор качества	Расшифровка
4	CONFIG_ERROR	Сигнал неправильно сконфигурирован
8	NOT_CONNECTED	Сигнал принят на обслуживание, но устройство еще не инициализировано
12	DEVICE_FAILURE	Ошибка инициализации устройства
20	LAST_KNOWN	КП нет на связи, последнее пришедшее значение
24	COMM_FAILURE	КП нет на связи
28	OUT_OF_SERVICE	Модуль не запущен
64	UNCERTAIN	Связь установлена, но значения еще не пришли
192	GOOD	Значение сигнала достоверно
216	LOCAL_OVERRIDE	Значение достоверно, введено вручную

Разрыв связи регистрируется, если за **Время ожидания ответа от станции** подчиненная станция не отвечает на запрос. При отсутствии ответа от подчиненной станции происходит повтор отправки запросов. По истечении максимального числа повторов связь с подчиненной станцией считается потерянной и выставляется плохое качество сигналов **COMM\_FAILURE**. При кратковременном разрыве связи станция успевает ответить за отведенное **Количество повторов отправки команд** (запросов) и качество сигналов при кратковременном разрыве связи не изменяется.





Модуль имеет возможность обрабатывать ошибки, чтобы не выставлять плохое качество данных при возникновении единичных сбоев в поведении устройства и чтобы сохранить целостную картину при резервных переходах устройств. Алгоритмы обработки ошибок следующие:

- если устройство при запросе данных возвращает строку **SLAVE DEVICE BUSY** (запись строки происходит в поле качество сигнала), то запрос данных не повторяется, связь со станцией не меняется, модуль переходит к запросу следующих данных. Недоставленные данные будут перезапрошены в следующем цикле опроса;
- если устройство при запросе данных возвращает строку **ILLEGAL DATA ADDRESS** (запись строки происходит в поле качество сигнала), либо другой код ошибки, то по данному запросу выставляется качество **OPC\_QUALITY\_BAD**;
- если подчиненная станция не отвечает и количество повторов отправки команд (запросов) достигло максимального, то сигналам по данному запросу выставляется качество **OPC\_QUALITY\_BAD**.

## 1.2.2. Метка времени

Протокол Modbus не оперирует понятием время возникновения данных, обычно сигналам присваивается время сервера на момент получения данных (свойство 4 (**TimeStamp**)).

Модуль Modbus RTU Master реализует расширение спецификации в части передачи метки времени. Метка времени может быть использована как для формирования в контроллере времени изменения значения, так и для формирования однозначной последовательности событий, возникающих на уровне контроллера, что невозможно в обычных условиях, так как опросчик может не успеть прочитать данные, если данные очень быстро изменяются.

Для передачи метки времени параметров модулем Modbus RTU Master используются протокольные типы с меткой времени \*\_TIME (например, TM2\_TIME, TMF4\_TIME и т.д.). Протокольные типы с меткой времени представлены логической структурой:

Рег./Поз.	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
1-4	Данные занимают от 1 до 4 регистров в зависимости от типа сигнала															
5	секунды								миллисекунды							
6	час								минуты							

Рег./Поз.	8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
7	год							месяц				день				

Для возможности взаимодействия модуля и контроллера на стороне контроллера следует поддерживать формирование описанной структуры.

Модуль, работая с сигналом, который имеет тип с меткой времени, формирует запрос на чтение с учетом типа сигнала и дополнительной структуры длиной три регистра. При получении данных метка времени, лежащая в дополнительной структуре, преобразуется к стандартному времени и записывается в свойство 4 (**TimeStamp**) одновременно с записью значения в свойство 2 (**Value**).

Для получения метки времени у подчиненной станции должен быть поддержан формат структур (таблица выше) по сигналам, которые должны содержать метку времени. Поля структуры должны заполняться на стороне подчиненной станции.

В SePlatform.Data Server для сигналов, значения которых должны передаваться с меткой времени, в карте адресов Modbus в значении **Метка времени** следует установить значение «Да» ([стр. 39](#)).

## 1.3. Стратегия формирования запросов на чтение

Модуль Modbus RTU Master предоставляет выбор варианта составления таблицы поллинга, т.е. возможность регулировки максимальной длины запроса на чтение и выбор варианта учитывать или не учитывать разрывы данных в адресном пространстве протокола.

Существуют два варианта формирования таблицы поллинга. Один из них позволяет оптимизировать количество запросов, другой - объем запрашиваемых данных. Вариант формирования таблицы поллинга указывается в параметре **Стратегия формирования запросов на чтение**:

- **«По максимуму»** - строится карта запросов максимальной длины, не учитывая фрагментацию запрашиваемых данных - объединение в один запрос расположенных не подряд данных. Например, если есть 2 сигнала с адресом «1» и «125», они объединяются в один запрос, а лишние данные при получении игнорируются (т.е. игнорируются данные с адресами от «2» до «124»). Таким образом, оптимизируется количество отправляемых запросов за счет увеличения трафика (количества переданной информации);
- **«На сплошные данные»** - строится карта запросов только по необходимым данным. Таким образом, оптимизируется трафик за счет потери времени. Потеря времени происходит из-за необходимости построения и отправки нескольких запросов.

Максимальная длина запроса равна «125» регистров. Длина запроса задается в параметрах конфигурации модуля. Сигналы с типами больше одного регистра отправляются и принимаются за один запрос независимо от настроек длины запроса.



### ПРИМЕР

#### Формирование карты поллинга

Если заданы адреса на чтение «1», «7»-«25», «47» и «130», то для стратегии формирования запросов:

- **«по максимуму»** - будут сформированы два запроса: один из запросов по адресам от «1» до «125», второй запрос от «126» до «130»;
- **«на сплошные данные»** - будет сформировано четыре запроса: по адресу «1», по адресам от «7» до «25», по адресу «47» и по адресу «130».

## 1.4. Отправка управляющих и регулирующих воздействий

Отправка команд телеуправления и телерегулирования выполняется модулем Modbus RTU Master только когда пара резервируемых серверов активна и SePlatform.Data Server находится в режиме РАБОТА.

Отправка управляющих воздействий имеет больший приоритет, чем опрос станций. При необходимости отправки управляющего воздействия модуль вставляет пакет, содержащий команду в начало очереди запросов на отправку подчиненной станции.

Модуль позволяет указать количество повторов отправки команд в случае неуспешной отправки. Повторы отправки команд происходят, если:

- устройство вернуло в ответ исключение с кодом **SLAVE DEVICE BUSY**. Если в итоге запись так и не была выполнена, то состояние связи выставляется в положение отсутствует;
- вышло время ожидания ответа от станции, т.е. произошел разрыв связи. После истечения максимального количества повторов отправки команд (запросов) сигналам выставляется плохое качество.

Качество данных является показателем достоверности информации. Сигналы управления плохого качества не отправляются.

В случае если подчиненная станция получает запрос, но не может его обработать (например, чтение несуществующей ячейки или регистра), то подчиненная станция возвращает ответ модулю с кодом ошибки.

Модуль имеет возможность отправлять все сигналы телерегулирования при проявлении подчиненной станции на связи. При этом может произойти отправка сигнала, содержащего устаревшую информацию, из-за этого функцию необходимо использовать осторожно.

Отправка сигнала телеуправления (ТС) может проводиться в сегменте **Holding Registers** адресного пространства. Для возможности подачи телеуправления при конфигурировании сигнала следует указать номер бита, который будет устанавливаться в «1» при отправке сигнала телеуправления. При подаче телеуправления модуль формирует регистр для записи в устройство таким образом, что все биты, кроме указанного в настройках адреса сигнала телеуправления, будут сброшены в ноль. Например, если в адресе сигнала параметру **Номер бита** установлено значение «5», то при отправке ТС в **Holding Registers** запишется число «00000000 00010000» или «32» в зависимости от настроек подчиненной станции.

### 1.4.1. Сигнал доставки команды

Для исходящих сигналов модуль Modbus RTU Master позволяет настраивать сигналы доставки ([стр. 40](#)) управляющих и регулирующих воздействий до подчиненной станции, которые формируются по правилам из таблицы:

Значение	Описание
2	Доставка подтверждена
1	Ожидание подтверждения доставки
-1	Нет связи со станцией
-2	Вышел таймаут подтверждения запроса
-1000 + код исключения	Устройство вернуло код исключения

## 1.5. Работа модуля в резерве

Модуль ведёт опрос подчиненных станций только в режиме РАБОТА. В режиме РЕЗЕРВ модуль не работает.

## 2. Настройка обмена данными со станцией

### 2.1. Исходные данные для настройки

Для настройки обмена данными между модулем Modbus RTU Master и станцией Modbus необходимы следующие исходные данные:

1. Информация о станции Modbus:
  - параметры COM-порта;
  - IP-адрес и порт шлюза (при обмене данными через шлюз TCP).
2. Информация о параметрах, значения которых требуется получать или отправлять на станцию Modbus:
  - номер станции;
  - сегмент памяти, в котором хранятся данные;
  - адрес элемента данных в сегменте памяти;
  - тип данных.

### 2.2. Пример для настройки обмена данными

Требуется получать от станции Modbus информацию о состоянии задвижки, подавать команды управления задвижкой, а также получать значение уровня по команде.

Параметры COM-порта станции Modbus:

- Имя - «COM1»;
- Скорость (бит/с) - «9600»;
- Чётность - «Нет»;
- Стоповые биты - «Один»;
- Управление потоком - «Нет».

Параметры шлюза Ethernet/COM для обмена данными через шлюз TCP:

- IP-адрес - «192.168.0.1»;
- Порт - «10000».

Информация о параметрах, значения которых требуется получать или отправлять:

Номер станции	Сегмент памяти	Адрес	Тип	Описание	Действие на стороне SePlatform.Data Server
1	Holding Registers	0	Uint2	Состояние задвижки	Периодическое получение значения
1	Holding Registers	6 бит 0	Bool	Команда на открытие задвижки	Отправка значения
1	Holding Registers	6 бит 1	Bool	Команда на закрытие задвижки	Отправка значения
1	Holding Registers	6 бит 2	Bool	Команда на останов движения задвижки	Отправка значения

Номер станции	Сегмент памяти	Адрес	Тип	Описание	Действие на стороне SePlatform.Data Server
1	Input Registers	4	Uint2	Значение уровня	Периодическое получение значения с интервалом «1000» мс

## 2.3. Настройка в SePlatform.Development Studio

Чтобы настроить обмен данными со станцией Modbus:

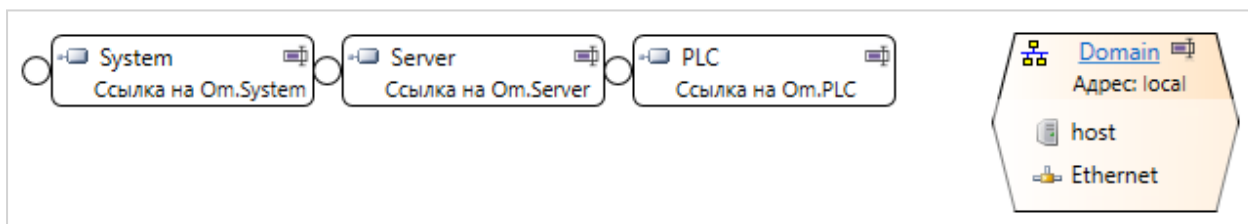
- добавьте в проект и настройте станцию Modbus RTU ([стр. 22](#));
- в SePlatform.Data Server добавьте и настройте Modbus RTU Master ([стр. 26](#));
- настройте канал для обмена данными со станцией ([стр. 32](#));
- настройте обмен данными между станцией и модулем Modbus RTU Master ([стр. 22](#));
- примените конфигурацию SePlatform.Data Server ([стр. 40](#)).



### ОБРАТИТЕ ВНИМАНИЕ

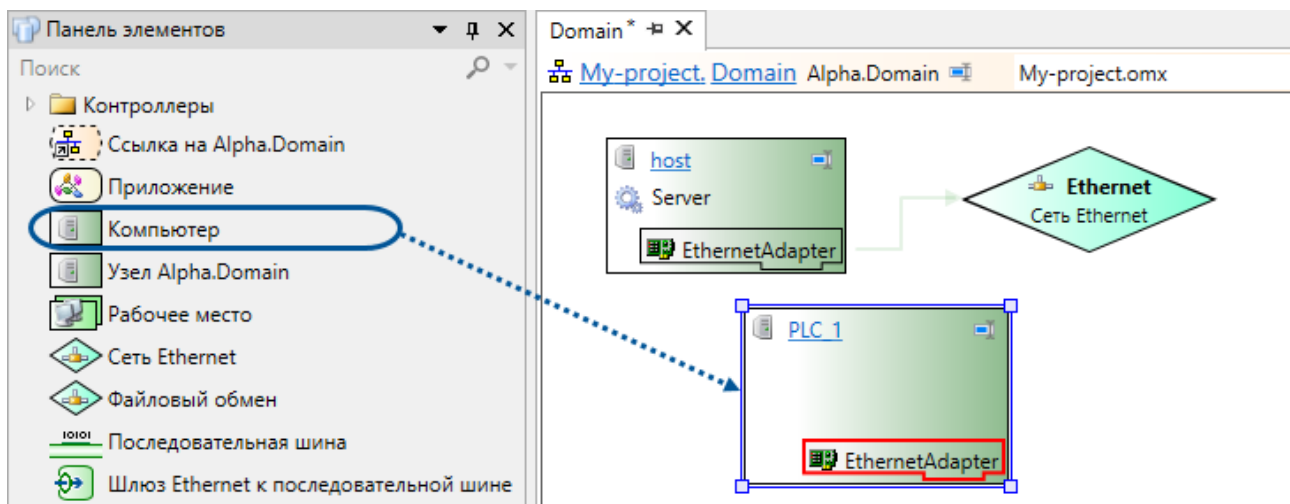
Порядок создания проекта и конфигурирование SePlatform.Data Server описаны в документации на SePlatform.Development Studio (раздел «Знакомство с SePlatform.Development Studio» руководства пользователя).

Далее приведено описание настройки обмена данными со станцией по протоколу Modbus уже сконфигурированного SePlatform.Data Server в существующем проекте.

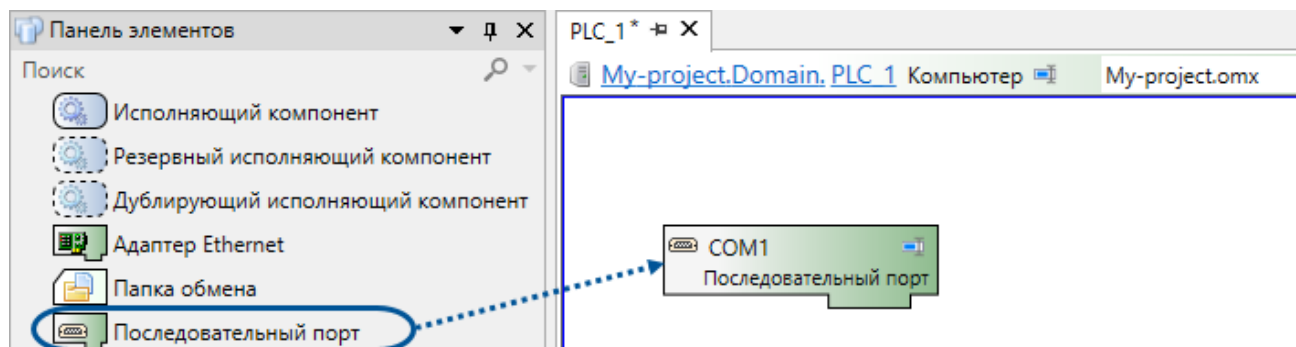


### 2.3.1. Настройка станции Modbus (ПЛК)

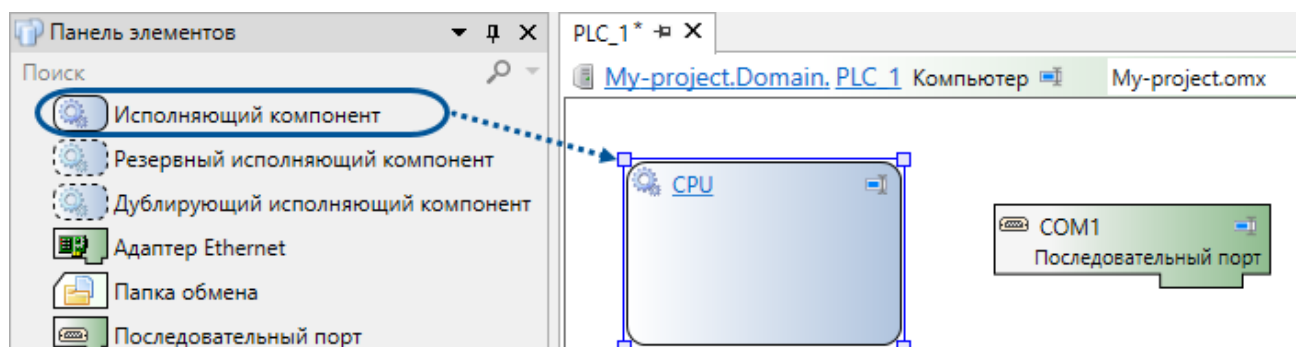
1. Перейдите в SePlatform.Domain и добавьте Компьютер. Задайте Имя, например, «PLC\_1».



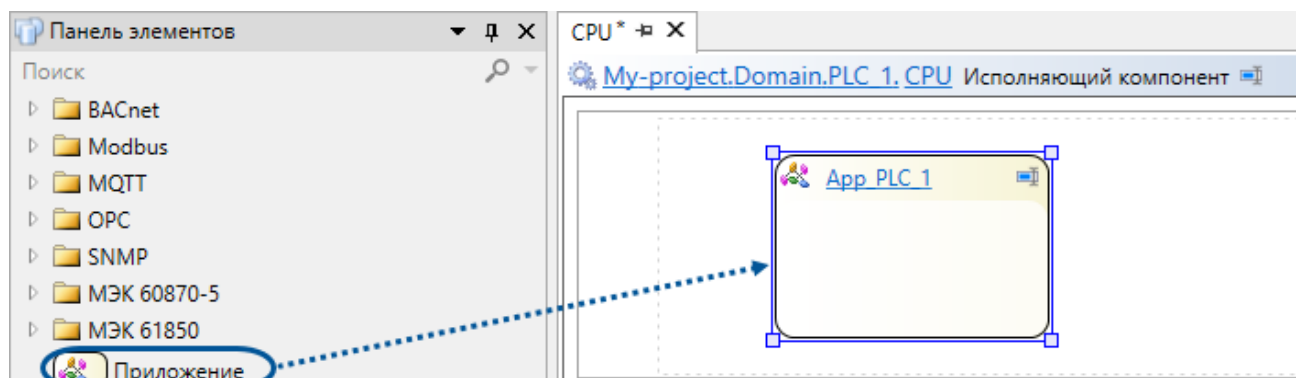
2. Перейдите в **Компьютер** и удалите **Адаптер Ethernet**, а вместо него добавьте **Последовательный порт**. Укажите **Имя**. В примере для настройки обмена данными имя COM-порта - «COM1» (стр. 21):



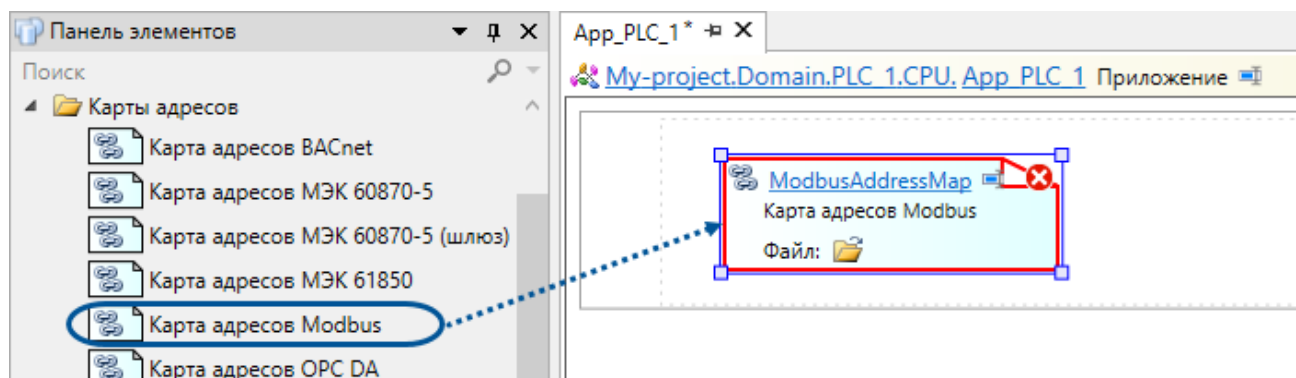
3. Добавьте **Исполняющий компонент** и задайте **Имя**, например, «CPU».



4. Перейдите в **Исполняющий компонент** и добавьте **Приложение**. Задайте **Имя**, например, «App\_PLC\_1».




5. Перейдите в **Приложение** и добавьте элемент **Карта адресов Modbus**.



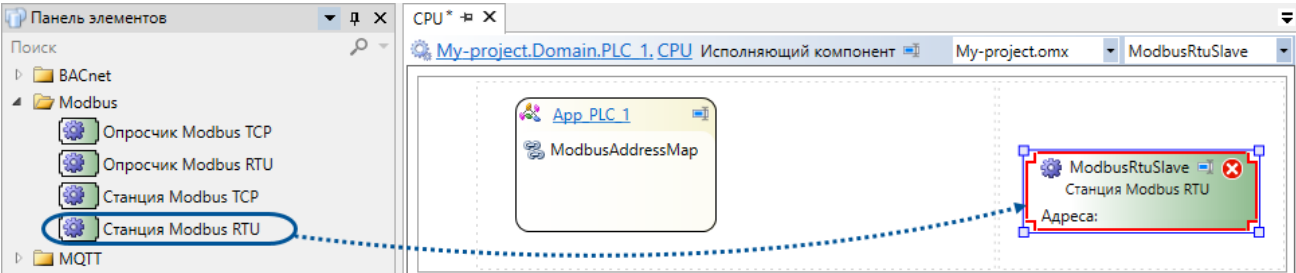
В дальнейшем в **Приложение** будут добавлены логические типы, с которыми будет взаимодействовать модуль Modbus RTU Master.

6. На элементе **Карта адресов Modbus**:

- нажмите кнопку  ;
- укажите место хранения карты адресов на диске;
- задайте имя карты адресов;
- нажмите кнопку **Открыть**.

Карта адресов будет создана. Если требуется открыть существующую карту адресов, то выберите файл карты адресов на диске и нажмите кнопку **Открыть**.

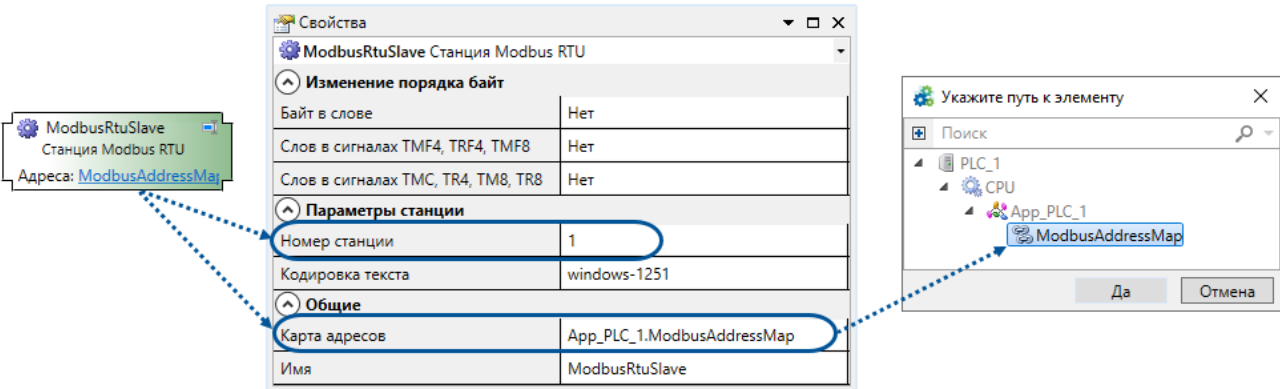
7. Вернитесь в **Исполняющий** компонент и добавьте **Станцию Modbus RTU**.




8. Настройте **Станцию Modbus RTU**:

- укажите **Номер** станции. В примере для настройки обмена данными номер станции - «1» ([стр. 21](#));
- в свойстве **Карта адресов** укажите ранее добавленную **Карту адресов Modbus**.

Значения остальных свойств можно оставить по умолчанию.












**ПРИМЕЧАНИЕ**

Если устройство Modbus, с которым настраивается обмен данными, на аппаратном уровне меняет местами байты в словах, то настройте свойства группы **Изменение порядка байт**.

Свойства Станции **Modbus RTU**:

Свойство	Описание
Изменение порядка байт	

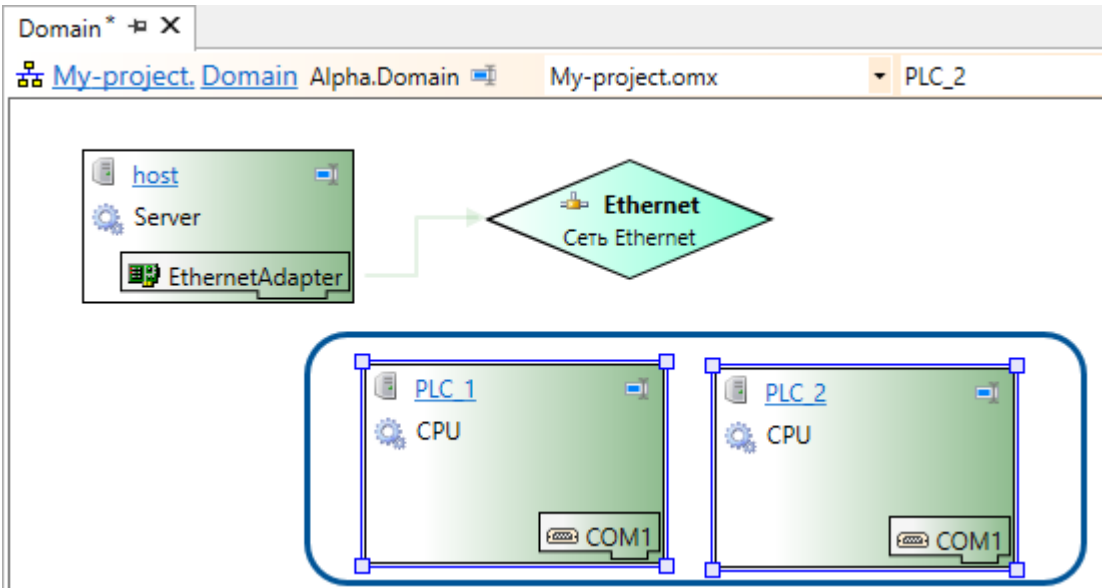


Свойство	Описание
Байт в слове	<p>Изменение порядка следования байт в регистре:</p> <ul style="list-style-type: none"> <li>➤ «Да» - байты регистра меняются местами:</li> </ul>  <ul style="list-style-type: none"> <li>➤ «Нет» - не изменять порядок следования байт.</li> </ul> <div>  <p><b>ОБРАТИТЕ ВНИМАНИЕ</b> Применяется ко всем протокольным типам, кроме TS и TC.</p> </div>
Слов в сигналах TMF4, TRF4, TMF8	<p>Изменение порядка следования слов в сигналах с протокольными типами TMF4, TMF8, TRF4:</p> <ul style="list-style-type: none"> <li>➤ «Да» - слова меняются местами:</li> </ul>  <p>Если параметру Байт в слове установлено значение «Да», то меняются местами слова и байты в словах:</p>  <ul style="list-style-type: none"> <li>➤ «Нет» - не изменять порядок следования слов.</li> </ul>
Слов в сигналах TMC, TR4, TM8, TR8	<p>Изменение порядка следования слов в сигналах с протокольными типами TMC, TR4, TM8, TR8:</p> <ul style="list-style-type: none"> <li>➤ «Да» - слова меняются местами:</li> </ul>  <p>Если параметру Байт в слове установлено значение «Да», то меняются местами слова и байты в словах:</p>  <ul style="list-style-type: none"> <li>➤ «Нет» - не изменять порядок следования слов.</li> </ul>
Параметры станции	
Номер станции	Номер станции Modbus RTU.
Кодировка текста	<p>Кодировка передаваемых текстовых данных. Значение по умолчанию «windows-1251».</p> <div>  <p><b>ОБРАТИТЕ ВНИМАНИЕ</b> В UTF-8 символы кириллицы и символы таблицы ASCII, выходящие за диапазон «0-127», занимают более «1» байта. Поэтому при настройке адреса сигнала для приёма/передачи строкового значения необходимо указывать достаточное количество регистров (параметр карты адресов <b>Размер строки</b>), иначе строка может приниматься/передаваться не полностью.</p> </div>

Свойство	Описание
Общие	
Карта адресов	Расположение карты адресов, которую будет использовать станция.
Имя	Название станции Modbus RTU в проекте.

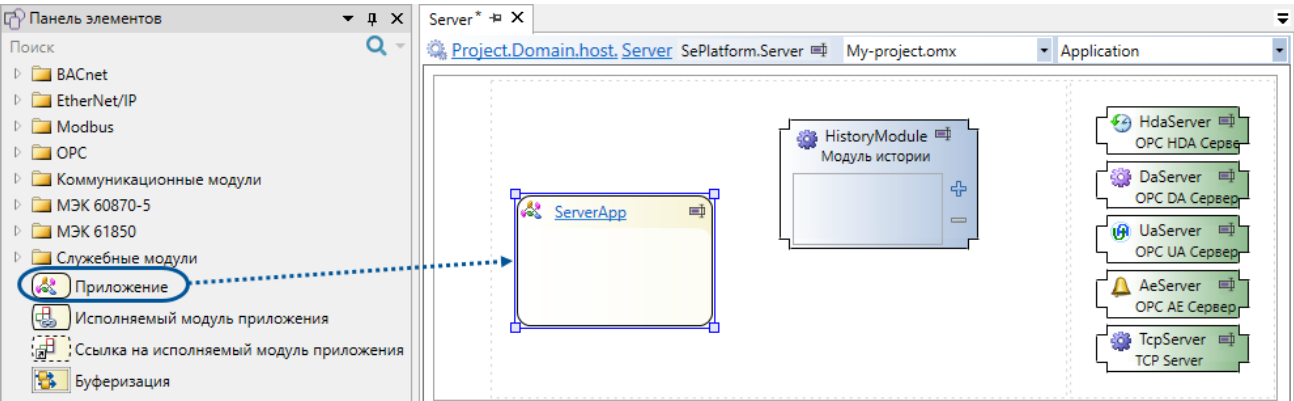
2.3.1.1. Настройка нескольких станций

В случае, если требуется настроить обмен данными с несколькими станциями Modbus, то в SePlatform.Domain добавьте необходимое количество Компьютеров и выполните настройку каждого их них (стр. 22).

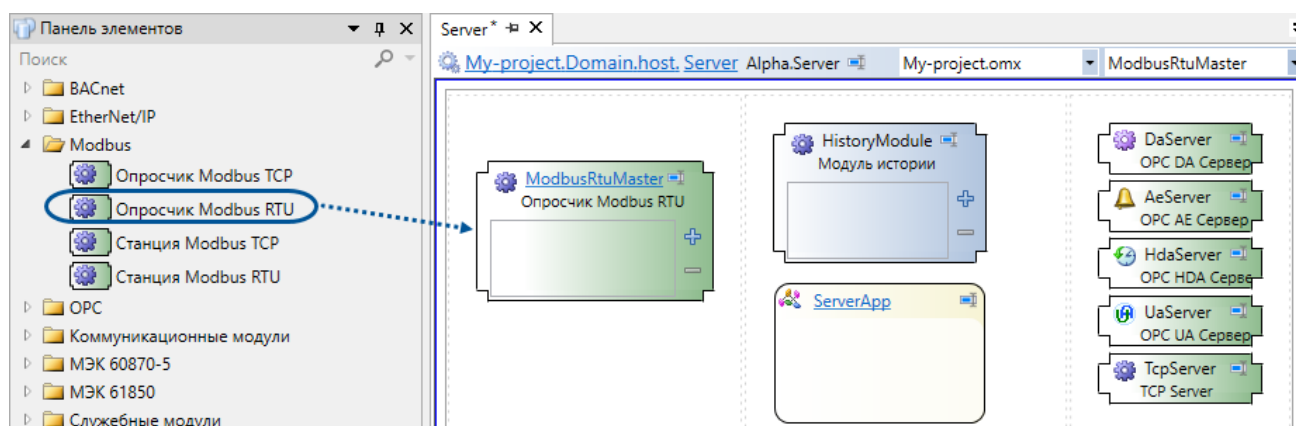


2.3.2. Настройка Modbus RTU Master

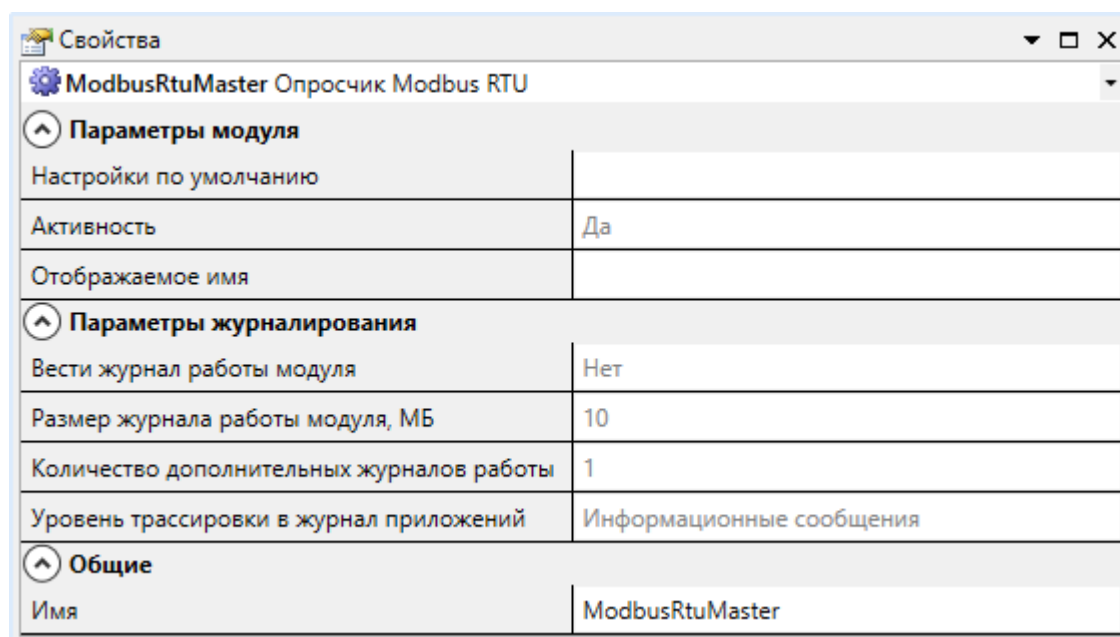
- 1. Перейдите в SePlatform.Domain.
- 2. Перейдите в узел домена «host», а затем в SePlatform.Data Server, и добавьте Приложение. Задайте Имя, например, «ServerApp».



- В дальнейшем в приложение «ServerApp» будут добавлены логические типы SePlatform.Data Server.
- 3. В SePlatform.Data Server добавьте Опросчик Modbus RTU.




Для свойств **Опросчика Modbus RTU** можно оставить значения по умолчанию.

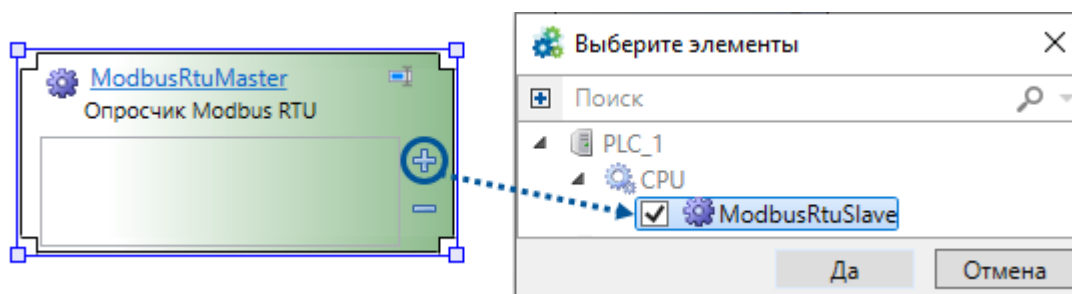


Свойство	Описание
Параметры модуля	
Настройки по умолчанию	Ссылка на элемент <b>Настройки по умолчанию</b> для задания однотипных настроек для нескольких одинаковых подчиненных станций.
Активность	Активность модуля при запуске/перезапуске SePlatform.Data Server: <ul style="list-style-type: none"> <li>➤ «Да» - модуль запущен;</li> <li>➤ «Нет» - модуль остановлен.</li> </ul> Управляется служебным сигналом «Active.Set».
Отображаемое имя	Название модуля, которое отображается в тегах служебных сигналов.
Параметры журналирования	

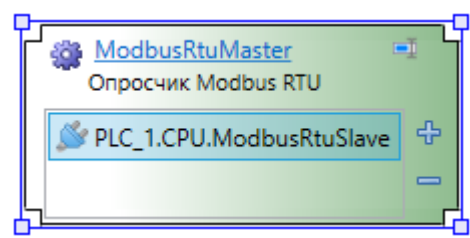
Свойство	Описание
Вести журнал работы модуля	<p>Ведение записи сообщений о работе модуля в журнал работы:</p> <ul style="list-style-type: none"> <li>➤ «Да» - вести журнал работы;</li> <li>➤ «Нет» - журнал работы не ведётся.</li> </ul> <p>Управляется служебным сигналом «FrameLogEnable.Set».</p>
Размер журнала работы модуля, МБ	Размер файла журнала работы модуля в мегабайтах. При достижении максимального размера создается новый файл, копия старого файла хранится на рабочем диске.
Количество дополнительных журналов работы	<p>Количество файлов заполненных журналов работы модуля:</p> <ul style="list-style-type: none"> <li>➤ минимальное количество - 1;</li> <li>➤ максимальное количество - 255.</li> </ul>
Уровень трассировки в журнал приложений	<p>Типы сообщений, которые выводятся в журнал приложений:</p> <ul style="list-style-type: none"> <li>➤ «Предупреждения и аварийные сообщения» - логические ошибки и ошибки работы модуля. Предупреждения содержат не критичные ошибки. Аварийные сообщения информируют об ошибках, которые влияют на работоспособность сервера;</li> <li>➤ «Информационные сообщения» - предупреждения и аварийные сообщения, а также основная информация о работе модуля;</li> <li>➤ «Отладочные сообщения» - предупреждения и аварийные сообщения, основная и детальная информация о работе модуля.</li> </ul> <p>Управляется служебным сигналом «SystemLogTraceLevel.Set».</p>
Общие	
Имя	Идентификатор модуля в конфигурации SePlatform.Data Server, значение сервисного сигнала «Id».

### 2.3.2.1. Настройка взаимодействия со станцией Modbus

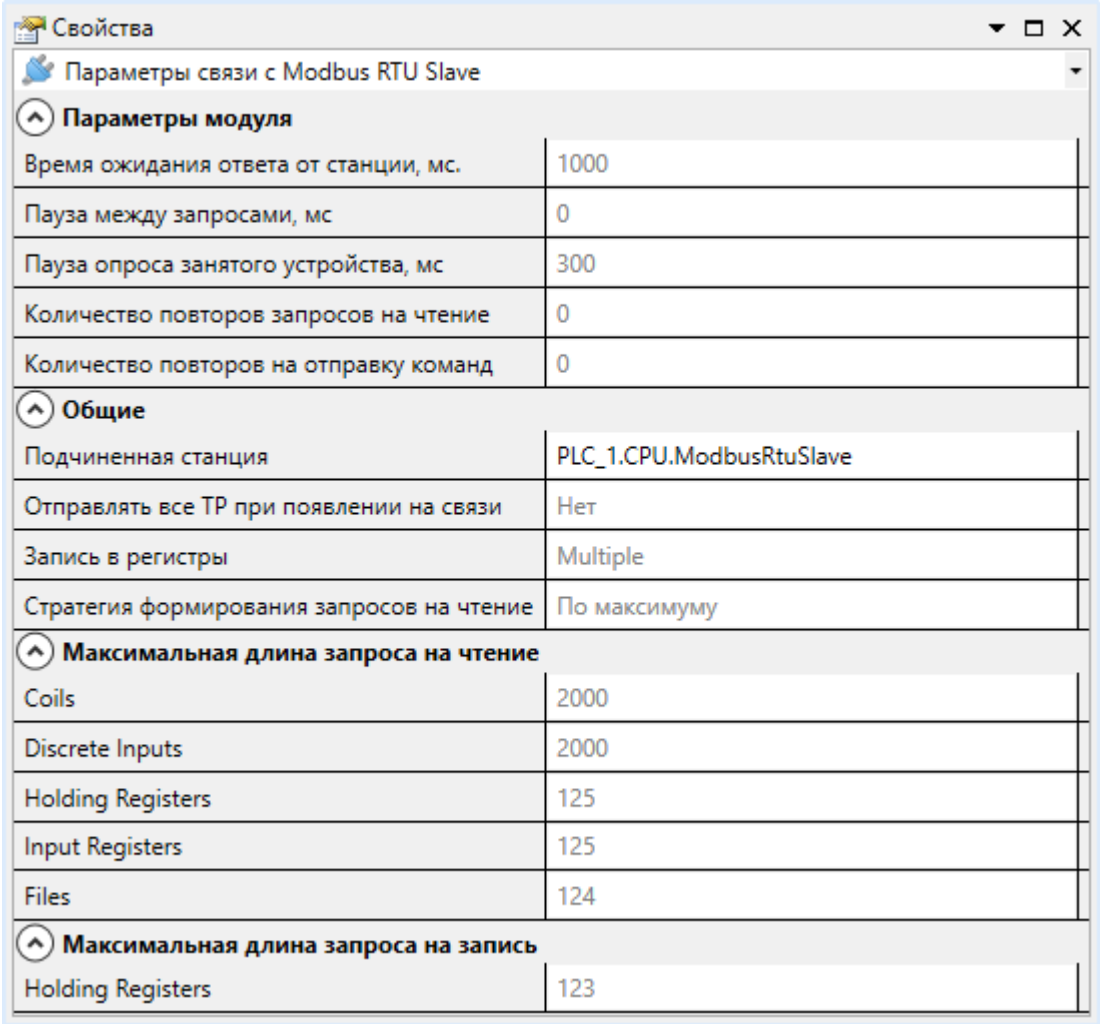
1. На **Опросчике Modbus RTU** нажмите кнопку  и укажите ранее настроенную Станцию **Modbus RTU**.



2. Чтобы перейти к настройкам параметров взаимодействия **Опросчика Modbus RTU** со Станцией **Modbus RTU**, выделите добавленную станцию.



Для всех параметров можно оставить значения по умолчанию.



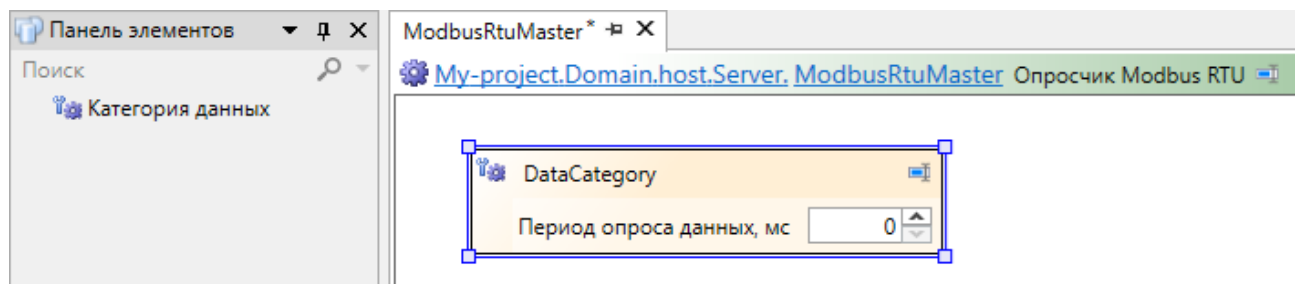
Свойство	Описание
Параметры модуля	
Время ожидания ответа от станции, мс	Время, в течение которого модуль ожидает ответ от станции. По истечении указанного времени связь со станцией считается потерянной. Значение по умолчанию - «1000» мс.
Пауза между запросами, мс	Интервал времени, выдерживаемый после получения ответа на запрос перед отправкой следующего запроса. Значение по умолчанию - «0» мс.

Свойство	Описание
Пауза опроса занятого устройства, мс	Интервал времени между попытками подключения модуля к станции, если станция занята ( <b>SLAVE DEVICE BUSY</b> ). Минимальное значение - «0» мс. Значение по умолчанию - «300» мс.
Количество повторов запросов на чтение	Количество повторно отправленных запросов на чтение данных подчиненной станции. Значение по умолчанию - «0».
Количество повторов на отправку команд	Количество повторно отправленных команд подчиненной станции. Значение по умолчанию - «0».
Общие	
Подчиненная станция	Расположение в проекте подчиненной станции.
Отправлять все ТР при появлении на связи	Отправка команд телерегулирования, содержащих начальные значения, при появлении подчиненной станции на связи: <ul style="list-style-type: none"> <li>➤ «Да» - отправлять команды телерегулирования;</li> <li>➤ «Нет» - не отправлять команды телерегулирования.</li> </ul>
Запись в регистры	Используемая функция записи значений в регистры: <ul style="list-style-type: none"> <li>➤ «Single» - запись значения в один регистр. Выполняется только для протокольного типа TR2, занимающего «2» байта (функция «0x06 Write Single Register»);</li> <li>➤ «Multiple» - запись значений в несколько регистров (функция «0x10 Write Multiple registers»).</li> </ul> Значение по умолчанию - «Multiple».
Стратегия формирования запросов на чтение	Метод формирования карты поллинга: <ul style="list-style-type: none"> <li>➤ «На сплошные данные» - строится карта запросов только по необходимым данным;</li> <li>➤ «По максимуму» - строится карта запросов максимальной длины, не учитывая фрагментацию необходимых данных.</li> </ul>
Максимальная длина запроса на чтение	
Coils	Битовые данные, максимальное количество отправляемых подчиненной станции ячеек сегмента памяти Coils (сигналов телеуправления) за один запрос. Диапазон значений от «1» до «2000». Значение по умолчанию - «2000».
Discrete Inputs	Битовые данные, максимальное количество запрашиваемых у подчиненной станции ячеек сегмента памяти Discrete Inputs (сигналов телесигнализации) за один запрос. Диапазон значений от «1» до «2000». Значение по умолчанию - «2000».

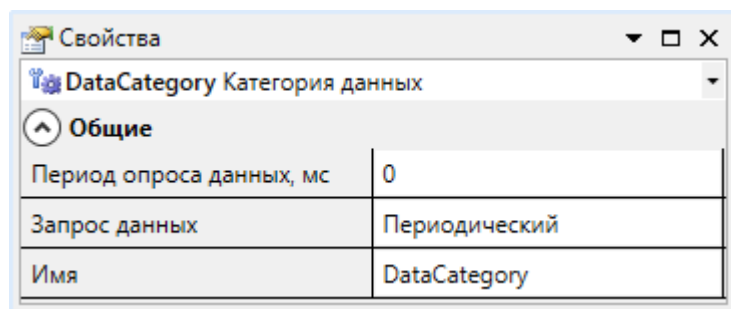
Свойство	Описание
<b>Holding Registers</b>	16-битные данные, максимальное количество отправляемых подчиненной станции регистров памяти Holding Registers (сигналов телерегулирования) за один запрос. Диапазон значений от «1» до «125». Значение по умолчанию - «125».
<b>Input Registers</b>	16-битные данные, максимальное количество запрашиваемых у подчиненной станции регистров памяти Input Registers (сигналов телеизмерения) за один запрос. Диапазон значений от «1» до «125». Значение по умолчанию - «125».
<b>Files</b>	16-битные данные, максимальное количество запрашиваемых у подчиненной станции регистров памяти Files за один запрос. Диапазон значений от «1» до «124». Значение по умолчанию - «124».
Максимальная длина запроса на запись	
<b>Holding Registers</b>	Максимальное количество 16-битных элементов данных для записи, отправляемых за один запрос. Диапазон значений от «1» до «123».

### 2.3.2.2. Настройка категорий данных


1. Перейдите в **Опросчик Modbus RTU** и добавьте элемент **Категория данных**.



2. Настройте свойства **Категории данных**.



Свойство	Описание
<b>Период опроса данных, мс</b>	Промежуток времени в миллисекундах, через который будут запрашиваться значения сигналов данной категории. Значение по умолчанию - «500».

Свойство	Описание
Запрос данных	<p>Способ запроса данных категории:</p> <ul style="list-style-type: none"> <li>➤ «Периодический» - запрос данных с частотой, указанной в параметре <b>Период опроса данных, мс</b>.</li> </ul> <div style="border: 1px solid #d4c000; padding: 10px; margin-top: 10px;"> <p> <b>ОБРАТИТЕ ВНИМАНИЕ</b></p> <p>Для категорий данных модуля Modbus RTU Master поддержан только «Периодический» способ запроса данных.</p> </div>
Имя	Имя группы опроса в конфигурации SePlatform.Data Server. Отображается в имени служебного сигнала для опроса группы.

3. Чтобы получать значение сигнала с определенным периодом, добавьте сигнал в соответствующую группу опроса. Для этого при заполнении карты адресов укажите для сигнала нужную **Категорию данных**.

В примере для настройки обмена данными требуется получать значение уровня с интервалом «1000» мс ([стр. 21](#)). Для этого настройте категорию данных со следующими параметрами:

- Имя - «Poll\_1000»;
- Запрос данных - «Периодический»;
- Период опроса данных, мс - «1000».

### 2.3.3. Настройка канала для обмена данными

Модуль Modbus RTU Master позволяет обмениваться данными со станцией Modbus по двум типам каналов:

- последовательный интерфейс RS-232/422/485;
- шлюз Ethernet/COM (Modbus RTU over TCP).

 **ОБРАТИТЕ ВНИМАНИЕ**

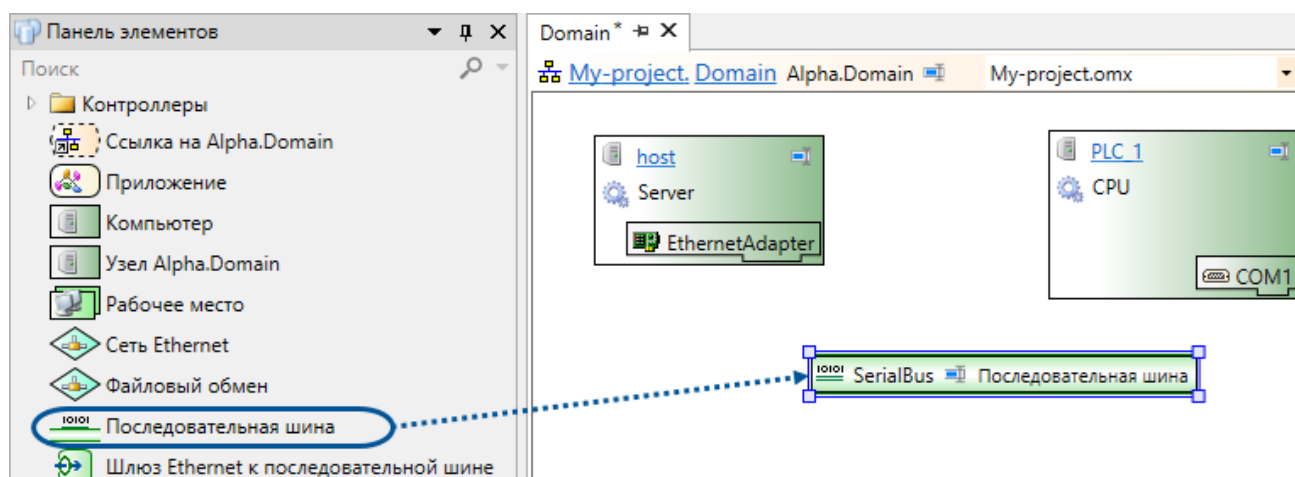
Модуль Modbus RTU Master одновременно может работать только по одному каналу.

#### 2.3.3.1. Последовательный интерфейс RS-232/422/485

Чтобы настроить обмен данными между модулем Modbus RTU Master и станцией Modbus через последовательный интерфейс RS-232/422/485:

1. Перейдите в **SePlatform.Domain** и удалите элемент **Сеть Ethernet**. Вместо него добавьте элемент **Последовательная шина**.



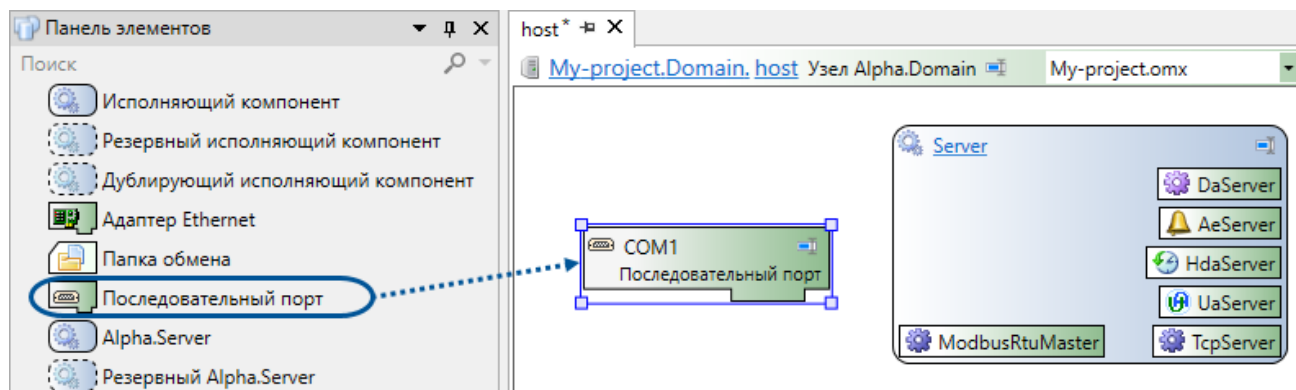


2. В свойствах **Последовательной** шины укажите параметры передачи данных, если они отличаются от значений, указанных по умолчанию.

Свойство	Описание
Передача данных	
Скорость бит/с	Скорость обмена данными между модулем Modbus RTU Master и станцией Modbus. Значение по умолчанию - «9600» бит/с.
Чётность	Выбор способа контроля четности для обнаружения ошибок при обмене данными: <ul style="list-style-type: none"> <li>➤ «Нет» - контроль четности не ведется, проверки ошибок нет;</li> <li>➤ «Чет» - проверка на четность;</li> <li>➤ «Нечет» - проверка на нечетность.</li> </ul> Значение по умолчанию - «Нет».
Стоповые биты	Количество стоповых битов, необходимых для правильного распознавания конца байта. Выбирается из выпадающего списка - «Один» или «Два» стоповых бита. Значение по умолчанию - «Один».
Управление потоком	Выбор режима управления потоком передаваемых данных: <ul style="list-style-type: none"> <li>➤ «Нет» - управление потоком выключено;</li> <li>➤ «Аппаратное» - аппаратная синхронизация, передатчик останавливает передачу данных, если приемник не готов к приему. Обеспечивает быструю реакцию на состояние приемника;</li> <li>➤ «Хоп/Хоф» - программная синхронизация, приемник посылает сигнал о неготовности передатчику, после чего передатчик останавливает посылку данных. Необходим двунаправленный канал связи.</li> </ul> Значение по умолчанию - «Нет».
Общие	
Имя	Имя последовательной шины в проекте.

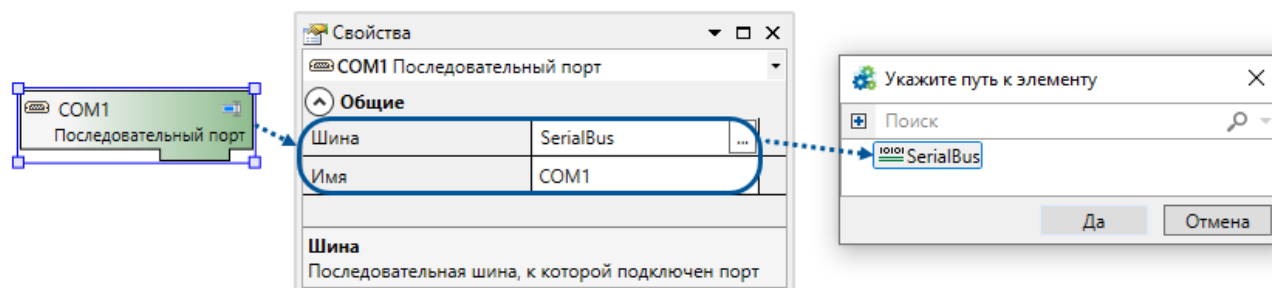
В примере для настройки обмена данными параметры передачи данных COM-порта не отличаются от значений по умолчанию, поэтому изменение значений свойств **Последовательной** шины не требуется ([стр. 21](#)).

3. Перейдите в узел домена «host» и удалите элемент **Адаптер Ethernet**. Вместо него добавьте элемент **Последовательный порт**.

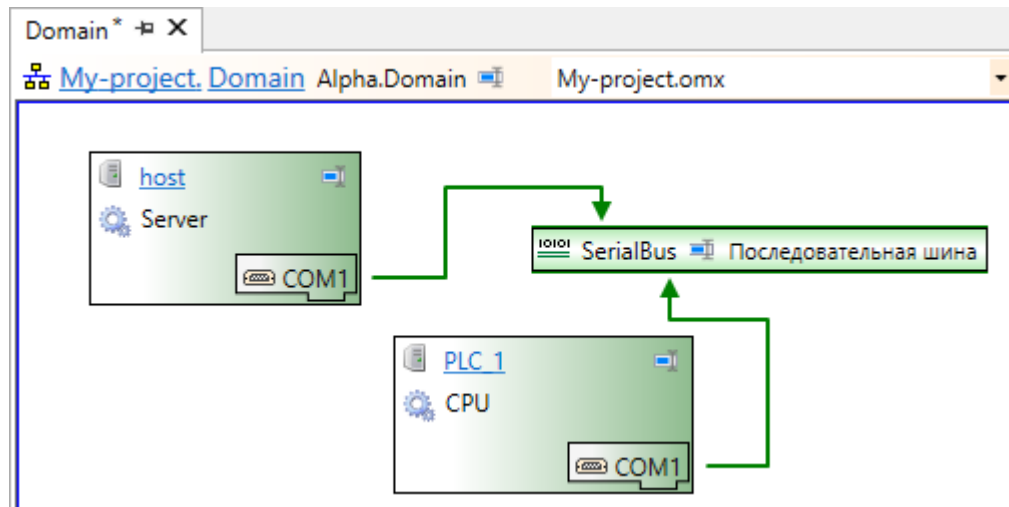


4. В свойствах **Последовательного порта** укажите:

- Шина - добавленная ранее **Последовательная шина**;
- Имя - системное имя COM-порта на компьютере с SePlatform.Data Server.



5. Вернитесь в SePlatform.Domain и соедините COM-порт станции Modbus с **Последовательной шиной**.

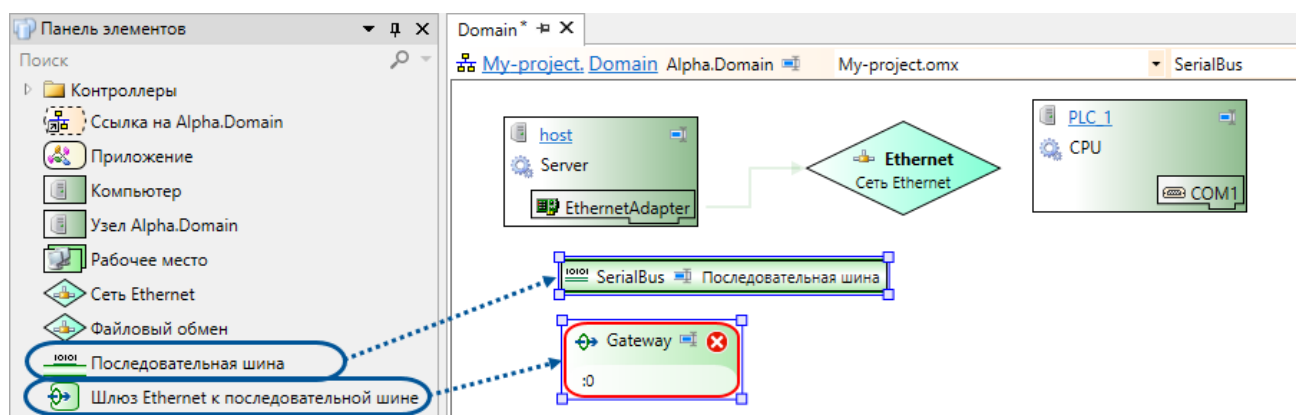


### 2.3.3.2. Шлюз Ethernet/COM (Modbus RTU over TCP)

Чтобы настроить обмен данными между модулем Modbus RTU Master и станцией Modbus через шлюз Ethernet/COM:

1. Перейдите в SePlatform.Domain и добавьте элементы:

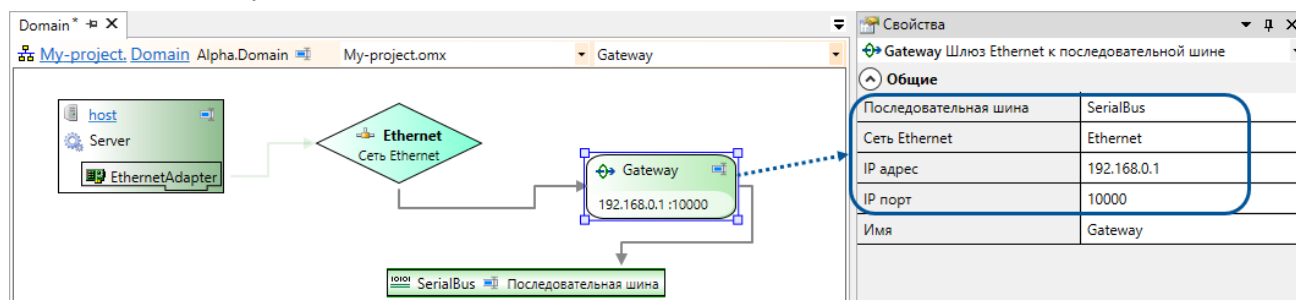
- Последовательная шина;
- Шлюз Ethernet к последовательной шине.



2. В свойствах **Последовательной** шины укажите параметры передачи данных, если они отличаются от значений, указанных по умолчанию ([стр. 33](#)).

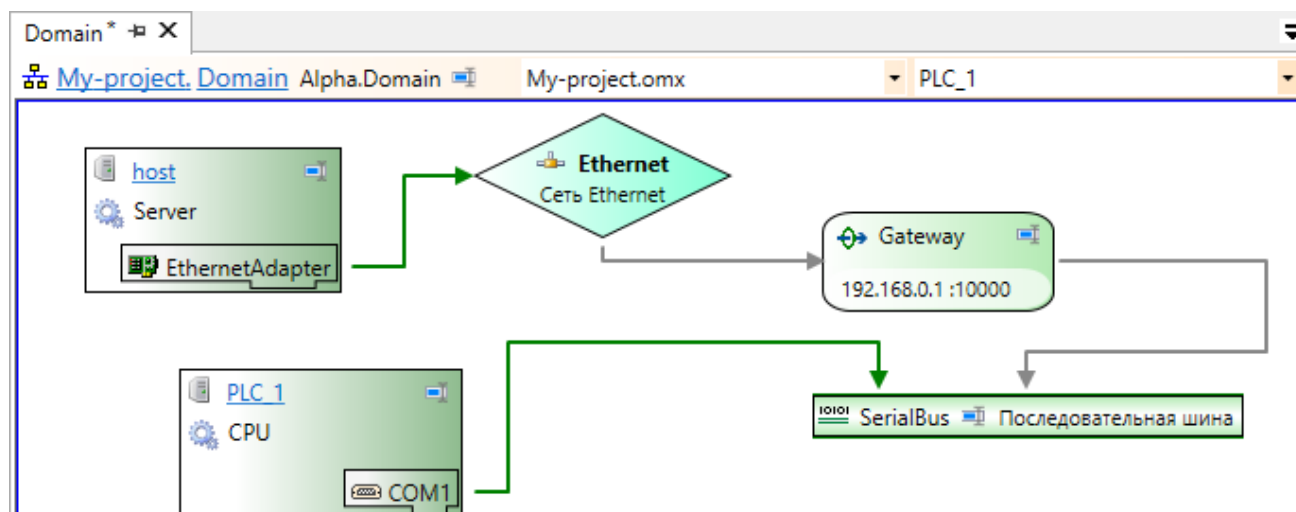
3. В свойствах **Шлюза Ethernet к последовательной шине** укажите:

- **Последовательная шина** - добавленная Последовательная шина;
- **Сеть Ethernet** - существующая сеть Ethernet;
- **IP адрес** - IP-адрес шлюза в сети Ethernet;
- **IP порт** - порт шлюза в сети Ethernet.



В примере для настройки обмена данными используются IP-адрес - «192.168.0.1» и порт «10000» ([стр. 21](#)).

4. Соедините COM-порт станции Modbus с **Последовательной** шиной.



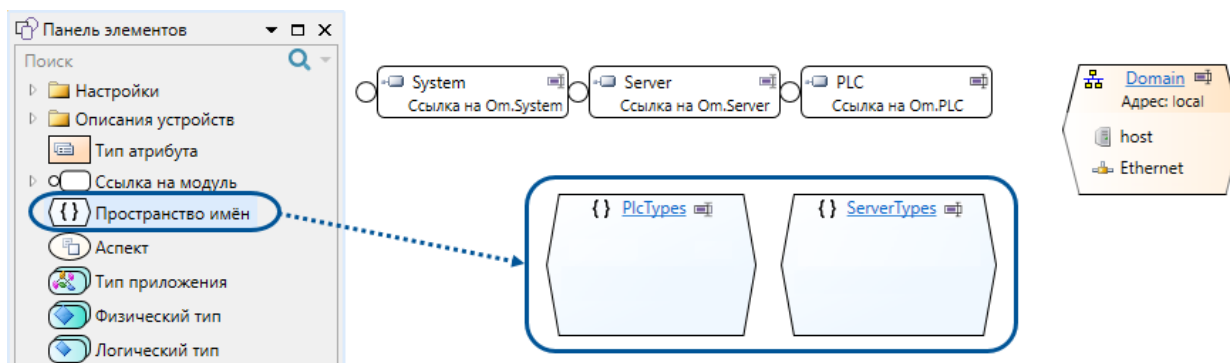
## 2.3.4. Настройка обмена данными

Для настройки обмена данными между модулем Modbus RTU Master и станцией Modbus необходимо:

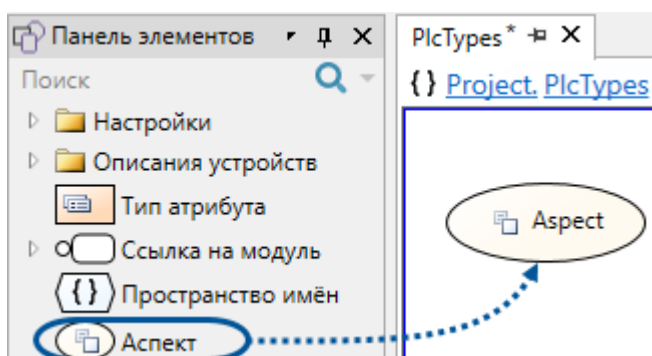
- настроить логические типы ПЛК и SePlatform.Data Server ([стр. 36](#));
- добавить соответствующие типы в приложения ПЛК и SePlatform.Data Server ([стр. 38](#));
- настройте получение и отправку значений сигналов ([стр. 39](#)).

### 2.3.4.1. Настройка логических типов

1. Добавьте в проект два элемента **Пространство имен**. Каждому элементу задайте **Имя**, например, «PlcTypes» и «ServerTypes». Это будут папки с логическими типами ПЛК и SePlatform.Data Server.



2. В каждое **Пространство имен** добавьте элемент **Аспект**.



Задайте **Имя** каждому добавленному аспекту:

- в пространстве имен «PlcTypes» - аспект «PLC»;
- в пространстве имен «ServerTypes» - аспект «Server».

### Логический тип ПЛК

1. Перейдите в **Пространство имен** «PlcTypes» и добавьте элемент **Логический тип**. В свойствах добавленного логического типа укажите:
  - **Имя** - имя логического типа ПЛК;
  - **Аспект** - добавленный ранее аспект «PLC».
2. Перейдите в добавленный **Логический тип** и опишите его структуру: добавьте сигналы необходимых типов, укажите им **Имя** и требуемое **Направление**.

Для получения/отправки значений сигналов из примера для настройки обмена данными добавьте логические типы ПЛК «Valve» и «Tank» со следующими сигналами: ([стр. 21](#))

Сигнал	Тип	Направление	Описание
Логический тип «Valve»			

Сигнал	Тип	Направление	Описание
«State»	Uint2	выход	Состояние задвижки
«Open»	Bool	вход	Команда на открытие задвижки
«Close»	Bool	вход	Команда на закрытие задвижки
«Stop»	Bool	вход	Команда на останов движения задвижки
Логический тип «Tank»			
«Level»	Int2	выход	Значение уровня

Логические типы ПЛК «Valve» и «Tank»:



## Логический тип SePlatform.Data Server

1. Перейдите в **Пространство имен «ServerTypes»** и добавьте элемент **Логический тип**. В свойствах добавленного логического типа укажите:

- **Имя** - имя логического типа SePlatform.Data Server;
- **Аспект** - добавленный ранее аспект «Server»;
- **Представляемый тип** - описанный ранее логический тип ПЛК.

Добавленный **Логический тип** будет представлять описанный ранее логический тип ПЛК, только со стороны SePlatform.Data Server.

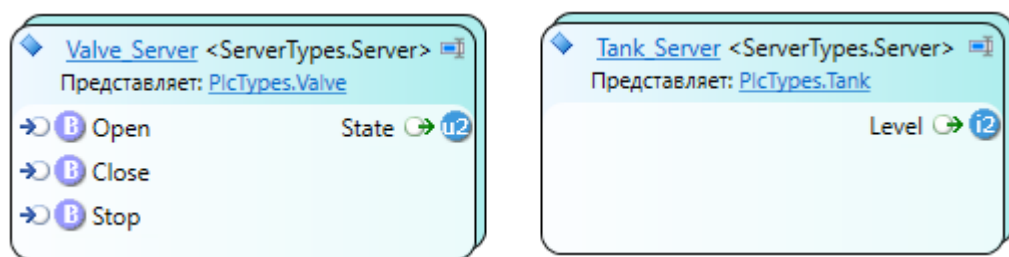
2. Перейдите в добавленный **Логический тип** и добавьте элемент **Ссылка**. В свойствах добавленной ссылки укажите:

- **Имя** - имя ссылки, например, «Reference\_Plc»;
- **Тип** - описанный ранее логический тип ПЛК.

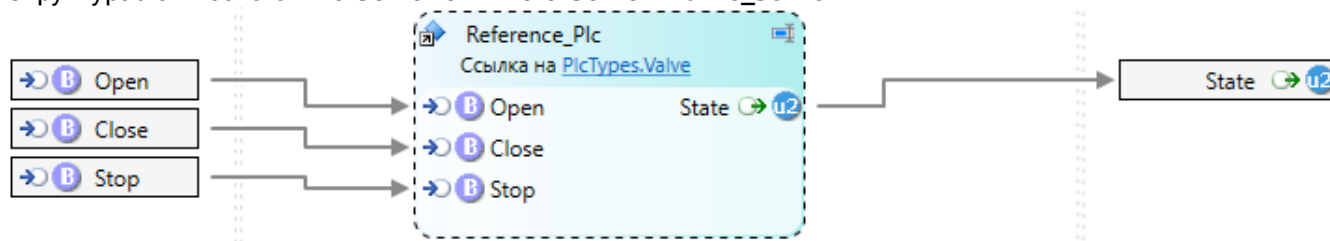
После указания **Типа** в элементе **Ссылка** отобразится структура логического типа ПЛК. Добавленная ссылка будет связывать логический тип SePlatform.Data Server с логическим типом ПЛК.

3. В контекстном меню ссылки выполните команду **Экспонировать входы и выходы**, в результате чего в логическом типе SePlatform.Data Server будут созданы сигналы с теми же именами и типами, что и в логическом типе ПЛК, а также прорисованы связи.

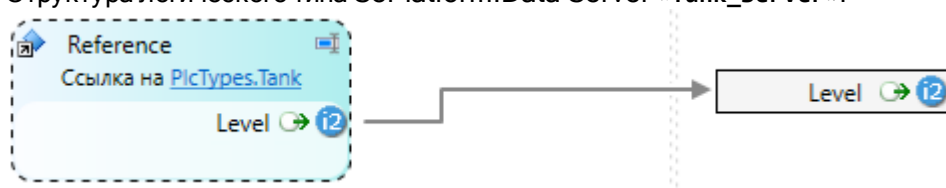
Логические типы SePlatform.Data Server «Valve\_Server» и «Tank\_Server» из примера для настройки обмена данными ([стр. 21](#)):



Структура логического типа SePlatform.Data Server «Valve\_Server»:

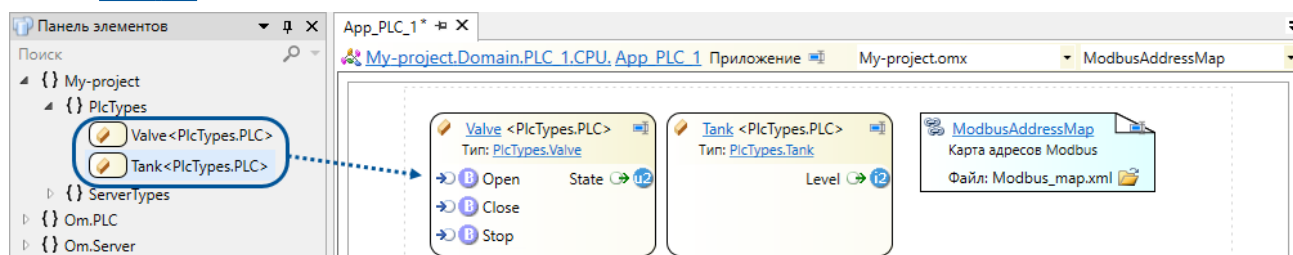


Структура логического типа SePlatform.Data Server «Tank\_Server»:



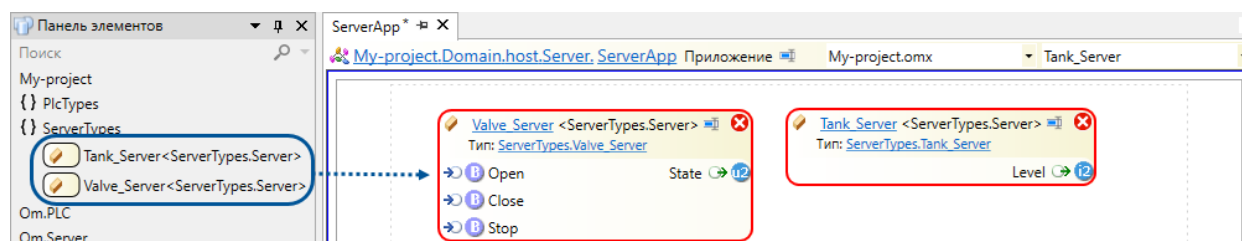
## Добавление типов в приложения ПЛК и SePlatform.Data Server

1. Перейдите в приложение ПЛК «App\_PLC\_1». Ранее в него уже был добавлен элемент Карта адресов Modbus (стр. 23). Теперь добавьте экземпляры логических типов ПЛК «Valve» и «Tank».



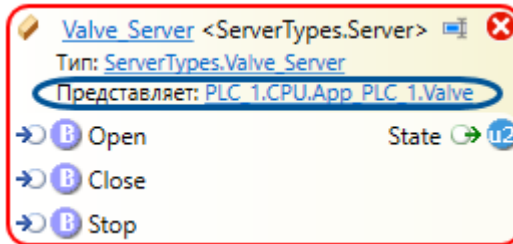
2. Перейдите в приложение SePlatform.Data Server «ServerApp»:

- 2.1. Добавьте экземпляры логических типов SePlatform.Data Server «Valve\_Server» и «Tank\_Server».

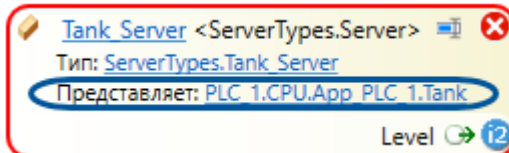


2.2. Добавленным типам на вкладке **Свойства** укажите **Представляемый объект** - экземпляр логического типа ПЛК, ранее добавленный в приложение «App\_PLC\_1»:


- для типа «Valve\_Server» - тип ПЛК «Valve»;



- для типа «Tank\_Server» - тип ПЛК «Tank».



2.3. Инициализируйте ссылки:

- вызовите контекстное меню и выполните команду **Инициализировать все ссылки**;
- нажмите кнопку  на панели инструментов или клавишу «F5» на клавиатуре.






## 2.3.4.2. Получение и отправка значений сигналов

Чтобы получать от ПЛК значения сигналов и отправлять в ПЛК команды управления перейдите в **Карту адресов Modbus** и настройте параметры адреса сигнала:

- **Привязка** - «непосредственно».
- **Сегмент** - сегмент памяти станции Modbus, в котором находится элемент данных.
- **Адрес** - адрес элемента данных в выбранном сегменте памяти, значение которого требуется получать или отправлять в ПЛК. Значение в диапазоне от «0» до «65535».
- **Номер бита** - номер бита, значение которого требуется получать или отправлять в ПЛК (только для сигналов типа Bool сегментов Input Registers и Holding Registers). Значения в диапазоне от «0» до «15».

Чтобы получать значения сигналов с определенной частотой, в параметре **Категория данных** укажите соответствующую категорию данных, настроенную в **Опроснике Modbus RTU**.

Заполненная карта адресов для сигналов из примера для настройки обмена данными ([стр. 21](#)):

My-project.Domain.PLC 1.CPU.App PLC 1, ModbusAddressMap Карта адресов Modbus										
	Сигнал	Тип	Привязка	Сегмент	Адрес	Номер бита	Номер записи в файле	Метка времени	Размер строки	Категория данных
	Valve.State	uint2	непосредственно	Holding Registers	0					
	Valve.Open	bool	непосредственно	Holding Registers	6	0				
	Valve.Close	bool	непосредственно	Holding Registers	6	1				
	Valve.Stop	bool	непосредственно	Holding Registers	6	2				
	Tank.Level	int2	непосредственно	Input Registers	4					Poll_1000

В зависимости от типа сигнала и указанного сегмента памяти могут быть доступны для настройки дополнительные параметры адреса:

- **Номер записи в файле** - номер записи в файле (для сегмента Files). Диапазон значений от «0» до «9999».
- **Метка времени** - передавать или получать значение сигнала с меткой времени ([стр. 17](#)).

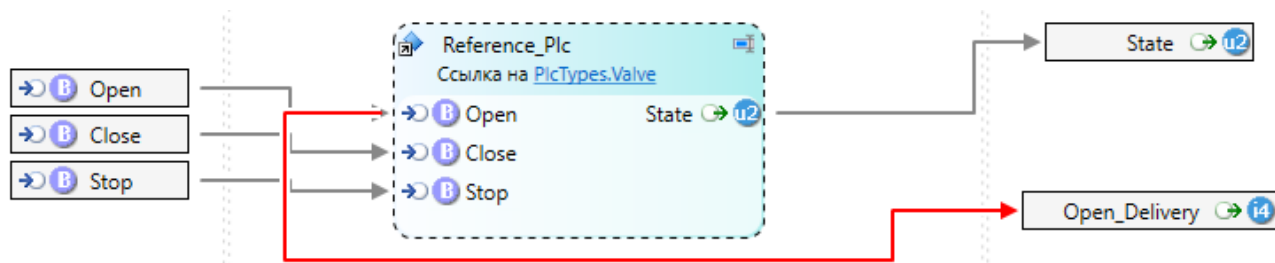
- **Размер строки** - размер строки в регистрах Modbus (для сигналов типа String). Диапазон значений от «1» до «123».

### 2.3.4.3. Настройка сигнала доставки

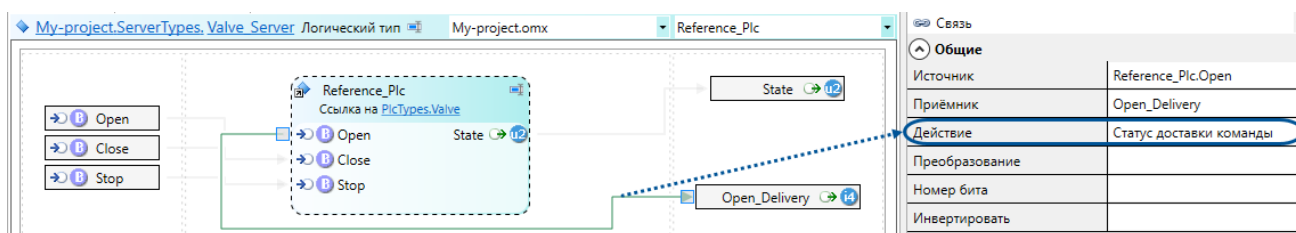
Для получения результата подачи команды в ПЛК следует настроить в логическом типе SePlatform.Data Server дополнительный сигнал типа Int4 на получение значения статуса доставки команды в ПЛК.

Чтобы получать статус доставки команды в ПЛК:

1. Перейдите в логический тип SePlatform.Data Server «Valve\_Server».
2. Добавьте сигнал типа Int4 для получения статуса доставки команды, например, «Open\_Delivery» для получения статуса доставки команды «Open».
3. Добавленному сигналу на вкладке **Свойства** укажите значение параметра **Направление** - «выход».
4. Соедините сигнал «Open» в элементе **Ссылка** с сигналом «Open\_Delivery». Соединительная линия связи будет прорисована красным цветом.

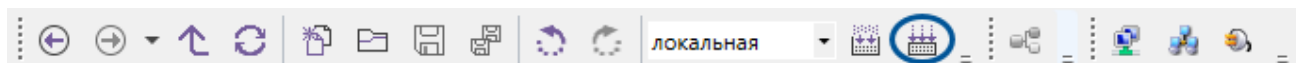


5. Выделите линию связи и на вкладке **Свойства** установите параметру **Действие** значение «Статус доставки команды». После этого линия будет прорисована зелёным цветом.

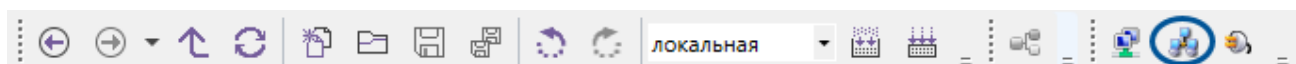


### 2.3.5. Применение конфигурации SePlatform.Data Server

1. Постройте решение. Конфигурация SePlatform.Data Server будет построена.



2. Перейдите в Мастер развёртывания.

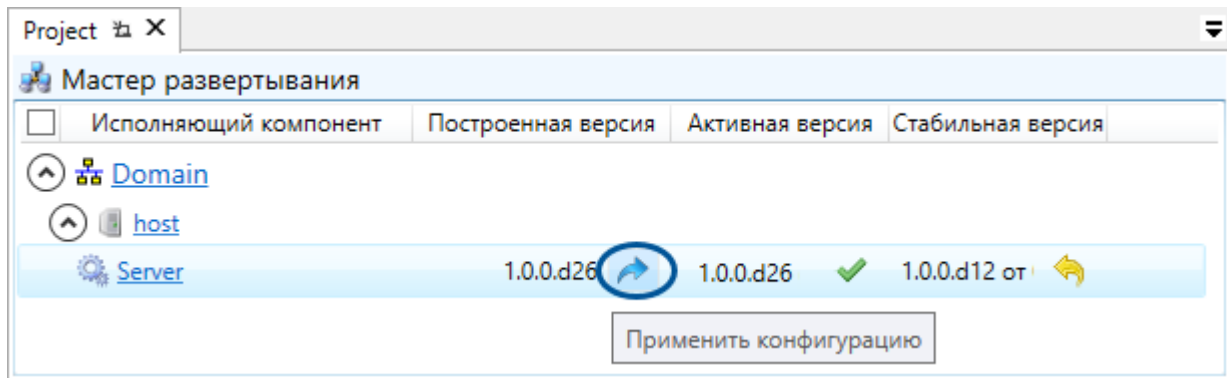


#### ОБРАТИТЕ ВНИМАНИЕ

Для успешного применения построенных конфигураций должен быть настроен SePlatform.Domain. Описание настройки приведено в документации на SePlatform.Domain (см. раздел «Конфигурирование» руководства администратора).



### 3. Примените конфигурацию к SePlatform.Data Server.



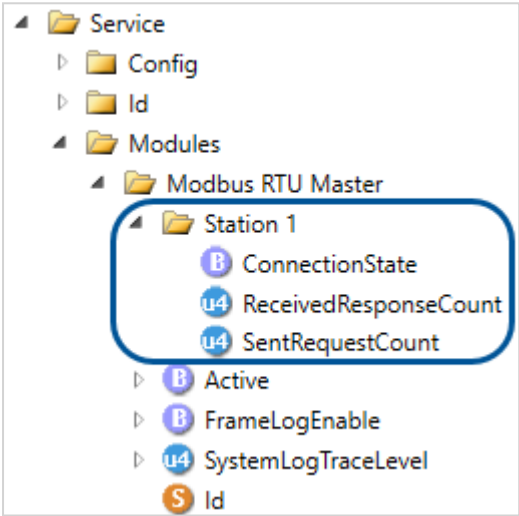
### 3. Диагностика работы модуля

#### 3.1. Служебные сигналы

Модуль Modbus RTU Master динамически создает для каждой подчиненной станции служебные сигналы для контроля состояния связи, количества запросов и ответов. Права доступа к служебным сигналам - только чтение.

Полный тег служебных сигналов имеет вид:

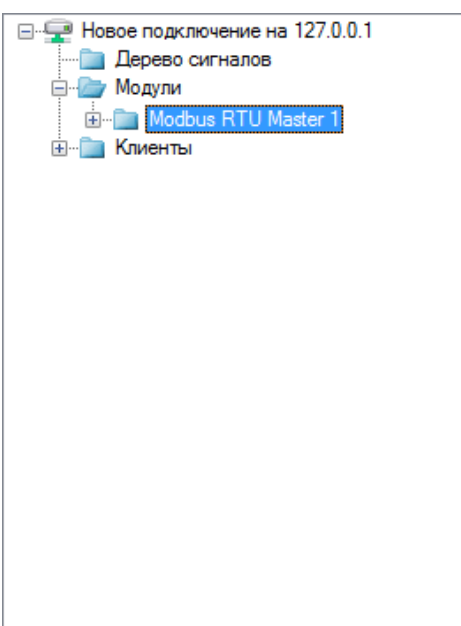
```
Service.Modules.<Имя модуля>.Station <номер станции>.<Имя сигнала>
```



Сигнал	Тип	Описание сигнала
«ConnectionState»	bool	Состояние связи со станцией: > «True» - есть связь; > «False» - нет связи.
«ReceivedResponseCount»	uint4	Количество принятых ответов.
«SentRequestCount»	uint4	Количество отправленных запросов.

#### 3.2. Статистика

Подробную информацию о работе модуля можно просмотреть с помощью сервисного приложения Статистика. Для просмотра параметров статистики модуля необходимо подключиться к SePlatform.Data Server и выбрать в дереве объектов модуль Modbus RTU Master.

	Имя	Значение
	<b>Общие параметры</b>	
	Идентификатор модуля	Modbus RTU Master 1
	Имя модуля	Modbus RTU Master 1
	Исполняемый файл	ModbusRTUMaster_Module.dll
	Версия	
	Активность	True
	Вести журнал работы	True
	Уровень детализации журнала работы	Информационные сообщения
	Предельный размер лога кадров	10
	Время старта	04.07.2023 15:38:43
	Лицензия	Основная - Да;
	<b>Очереди данных</b>	
	Размер очереди исходящих данных	0
	Размер очереди входящих данных	0
	<b>Канал</b>	
	Количество отправленных запросов	0
	Количество принятых ответов	0

Общая статистическая информация модуля представлена в группе **Общие параметры**.

Статистическая информация модуля о размере очереди данных представлена в группе **Очередь данных**:

Параметр	Описание
Размер очереди исходящих данных	Количество пакетов входящих данных, ждущих обработки.
Размер очереди входящих данных	Количество пакетов исходящих данных - команд, ждущих своей отправки.

Статистическая информация модуля о количестве пакетов данных представлена в группе **Канал**:

Параметр	Описание
Количество отправленных запросов	Общее количество отправленных пакетов данных.
Количество принятых ответов	Общее количество принятых пакетов данных.

Статистическая информация модуля о количестве обслуживаемых сигналов представлена в группе **Обслуживаемые сигналы** (рисунок ниже):

Параметр	Описание
Общее количество обслуживаемых сигналов	Суммарное количество обслуживаемых сигналов с разными типами данных.

По каждому протокольному типу данных предоставляется статистическая информация в виде количества обслуживаемых сигналов ([стр. 1](#)).

<div> <div>Новое подключение на 127.0.0.1</div> <ul style="list-style-type: none"> <li>Дерево сигналов           <ul style="list-style-type: none"> <li>Модули               <ul style="list-style-type: none"> <li><b>Modbus RTU Master 1</b></li> <li>Клиенты</li> </ul> </li> </ul> </li> </ul> </div>	<table> <thead> <tr> <th>Имя</th><th>Значение</th></tr> </thead> <tbody> <tr><td colspan="2"><b>Обслуживаемые сигналы</b></td></tr> <tr><td>Общее количество обслуживаемых сигналов</td><td>6</td></tr> <tr><td>TS</td><td>3</td></tr> <tr><td>TC</td><td>1</td></tr> <tr><td>TM2</td><td>1</td></tr> <tr><td>TMF4</td><td>0</td></tr> <tr><td>TMC</td><td>0</td></tr> <tr><td>TR2</td><td>0</td></tr> <tr><td>TRF4</td><td>0</td></tr> <tr><td>TR4</td><td>0</td></tr> <tr><td>STR</td><td>0</td></tr> <tr><td>STR-COMMAND</td><td>0</td></tr> <tr><td>TM2_Time</td><td>0</td></tr> <tr><td>TMF4_Time</td><td>0</td></tr> <tr><td>TMC_Time</td><td>0</td></tr> <tr><td>TR2_Time</td><td>0</td></tr> <tr><td>TR4_Time</td><td>0</td></tr> <tr><td>TRF4_Time</td><td>0</td></tr> <tr><td>Сигналы статуса доставки</td><td>1</td></tr> </tbody> </table>	Имя	Значение	<b>Обслуживаемые сигналы</b>		Общее количество обслуживаемых сигналов	6	TS	3	TC	1	TM2	1	TMF4	0	TMC	0	TR2	0	TRF4	0	TR4	0	STR	0	STR-COMMAND	0	TM2_Time	0	TMF4_Time	0	TMC_Time	0	TR2_Time	0	TR4_Time	0	TRF4_Time	0	Сигналы статуса доставки	1
Имя	Значение																																								
<b>Обслуживаемые сигналы</b>																																									
Общее количество обслуживаемых сигналов	6																																								
TS	3																																								
TC	1																																								
TM2	1																																								
TMF4	0																																								
TMC	0																																								
TR2	0																																								
TRF4	0																																								
TR4	0																																								
STR	0																																								
STR-COMMAND	0																																								
TM2_Time	0																																								
TMF4_Time	0																																								
TMC_Time	0																																								
TR2_Time	0																																								
TR4_Time	0																																								
TRF4_Time	0																																								
Сигналы статуса доставки	1																																								

Статистическая информация в разделе нагрузка представляет собой количество изменений значений сигналов по разным типам сигналов - количество перезаписей значений сигналов. Перезаписи протокольных типов сигналов разделены на группы **Исходящие изменения** и **Входящие изменения**.

<div> <div>Новое подключение на 127.0.0.1</div> <ul style="list-style-type: none"> <li>Дерево сигналов           <ul style="list-style-type: none"> <li>Модули               <ul style="list-style-type: none"> <li>Modbus RTU Master 1                   <ul style="list-style-type: none"> <li><b>Нагрузка</b></li> <li>Список станций</li> </ul> </li> <li>Клиенты</li> </ul> </li> </ul> </li> </ul> </div>	<table> <thead> <tr> <th>Имя</th><th>Значение</th></tr> </thead> <tbody> <tr><td>Общее количество изменений</td><td>0</td></tr> <tr><td>TC</td><td>0</td></tr> <tr><td>TR2</td><td>0</td></tr> <tr><td>TRF4</td><td>0</td></tr> <tr><td>TR4</td><td>0</td></tr> <tr><td>STR-COMMAND</td><td>0</td></tr> <tr><td>TR2_Time</td><td>0</td></tr> <tr><td>TR4_Time</td><td>0</td></tr> <tr><td>TRF4_Time</td><td>0</td></tr> <tr><td colspan="2"><b>Входящие изменения</b></td></tr> <tr><td>Общее количество изменений</td><td>4</td></tr> <tr><td>TS</td><td>3</td></tr> <tr><td>TM2</td><td>1</td></tr> <tr><td>TMF4</td><td>0</td></tr> <tr><td>TMC</td><td>0</td></tr> <tr><td>STR</td><td>0</td></tr> <tr><td>TM2_Time</td><td>0</td></tr> <tr><td>TMF4_Time</td><td>0</td></tr> <tr><td>TMC_Time</td><td>0</td></tr> </tbody> </table>	Имя	Значение	Общее количество изменений	0	TC	0	TR2	0	TRF4	0	TR4	0	STR-COMMAND	0	TR2_Time	0	TR4_Time	0	TRF4_Time	0	<b>Входящие изменения</b>		Общее количество изменений	4	TS	3	TM2	1	TMF4	0	TMC	0	STR	0	TM2_Time	0	TMF4_Time	0	TMC_Time	0
Имя	Значение																																								
Общее количество изменений	0																																								
TC	0																																								
TR2	0																																								
TRF4	0																																								
TR4	0																																								
STR-COMMAND	0																																								
TR2_Time	0																																								
TR4_Time	0																																								
TRF4_Time	0																																								
<b>Входящие изменения</b>																																									
Общее количество изменений	4																																								
TS	3																																								
TM2	1																																								
TMF4	0																																								
TMC	0																																								
STR	0																																								
TM2_Time	0																																								
TMF4_Time	0																																								
TMC_Time	0																																								

Статистическая информация также выдается для каждой станции в отдельности. И содержит в себе ту же информацию только для своей станции.

### 3.3. Журнал работы модуля

Модуль Modbus RTU Master ведёт журнал работы, в который записывается информация о работе модуля и обмене данными с подчиненными станциями.



#### ОБРАТИТЕ ВНИМАНИЕ

Чтобы Modbus RTU Master вёл журнал работы, в общих параметрах модуля установите параметру **Вести журнал работы модуля** значение **Да** или установите сервисному сигналу модуля «FrameLogEnable.Set» значение «true».

Журнал работы модуля сохраняется в файл <имя модуля>.aplog в папке сервера, по умолчанию:

- в ОС Windows в папке C:\Program Files\SePlatform\SePlatform.Server\Logs;
- в системах Linux в директории /opt/SePlatform/Logs.

Для просмотра журнала работы модуля используется сервисное приложение Просмотрщик лога кадров.

№	Дата	Время	Описание	Начальный адрес	Количество
2	03.07.2023	15:45:03:213	Порт '\\.\COM2' открыт	26	3
3	03.07.2023	15:45:03:213	Подключаемся к станции		
4	03.07.2023	15:45:03:213	Исходящий кадр. Функция 3 (Read Holding Registers)		
5	03.07.2023	15:45:03:345	Входящий кадр. Функция 3 (Read Holding Registers)		
6	03.07.2023	15:45:03:345	Станция 1 подключена		
7	03.07.2023	15:45:13:413	Исходящий кадр. Функция 3 (Read Holding Registers)		
8	03.07.2023	15:45:13:545	Входящий кадр. Ошибка. Код 192. Функция 3 (Read Holding Re...		
9	03.07.2023	15:45:23:613	Исходящий кадр. Функция 3 (Read Holding Registers)		
10	03.07.2023	15:45:23:745	Входящий кадр. Функция 3 (Read Holding Registers)		
11	03.07.2023	15:45:33:813	Исходящий кадр. Функция 3 (Read Holding Registers)		
12	03.07.2023	15:45:33:945	Входящий кадр. Ошибка. Код 192. Функция 3 (Read Holding Re...		

Исходящий кадр. Функция 3 (Read Holding Registers)
--

0	1	2	3	4	5	6
0000	01	03	00	1A	00	03
						24

### 3.3.1. Структура кадров модуля

Исходящий и входящий кадры модуля Modbus RTU Master имеют структуру:

Адрес	Код функции	Данные	Контрольная сумма
1 байт	1 байт	до 253 байт	2 байта

- **Адрес** - номер подчиненной станции в диапазоне от 1 до 255, с которой выполняется обмен данными.
- **Код функции** - функция записи или чтения данных сегмента памяти подчиненной станции.
- **Данные** (формат и длина поля зависит от кода функции):
  - в исходящем кадре - данные, необходимые для выполнения функции;
  - во входящем кадре - данные, предоставленные подчиненной станцией.
- **Контрольная сумма** - проверка отсутствия ошибок в кадре.

Побайтовое представление и данные кадров модуля отображаются в соответствующих полях окна сервисного приложения Просмотрщик лога кадров.

Данные кадра	Начальный адрес	Количество									
	1	2									
Запись журнала	Исходящий кадр. функция 4 (Read Input Registers)										
Побайтовое представление в шестнадцатиричном формате		0	1	2	3	4	5	6	7	8	9
	0000	01	04	00	01	00	02	20	0B		

Индикация байт в поле побайтового представления кадра данных приведена в таблице:

Байт	Цвет	Описание
0	Голубой	Адрес подчиненной станции
1	Светло голубой	Код функции
от 2 до N	Светло-серый	Данные (формат и длина зависит от кода функции)
N+1, N+2	Желтый	Контрольная сумма

### 3.3.1.1. Кадры функций чтения

Исходящий кадр для функций «01 (0x01)» - «04 (0x04)» имеет структуру:

Адрес	Код функции	Начальный адрес	Количество	Контрольная сумма
1 байт	1 байт	2 байта	2 байта	2 байта

- **Начальный адрес** - адрес первого элемента сегмента памяти подчиненной станции, значение которого требуется прочитать.
- **Количество** - количество считываемых элементов.

Входящий кадр для функций «01 (0x01)» - «04 (0x04)» имеет структуру:

Адрес	Код функции	Число байт	Данные	Контрольная сумма
1 байт	1 байт	1 байт	N байт	2 байта

- **Число байт** - количество байт полученных данных.
- **Данные** - полученные значения.



## ПРИМЕР

Исходящий кадр функции «01 (0x01) Read Coils»:

Начальный адрес	Количество
104	3
Исходящий кадр. функция 1 (Read Coils)	
	0 1 2 3 4 5 6 7 8 9
0000	01 01 00 68 00 03 FD D7

Байт	Данные	Описание
0	01	Адрес станции - 1
1	01	Код функции - «01 (0x01)»
2, 3	00 68	Начальный адрес - 104
4, 5	00 03	Количество - 3
6, 7	FD D7	Контрольная сумма

Входящий кадр функции «01 (0x01) Read Coils»:

Адрес	Значение
104	False
105	False
106	False
Входящий кадр. функция 1 (Read Coils)	
	0 1 2 3 4 5 6 7 8 9
0000	01 01 01 00 51 88

Байт	Данные	Описание
0	01	Адрес станции - 1
1	01	Код функции - «01 (0x01)»
2	01	Число байт - 1
3	00	Значение - 0
4, 5	51 88	Контрольная сумма

Исходящий кадр функции «20 (0x14)» имеет структуру:

Адрес	Код функции	Число байт	Подзапрос				Контрольная сумма
			Тип ссылки	Номер файла	Номер записи	Длина записи	
1 байт	1 байт	1 байт	1 байт	2 байта	2 байта	2 байта	2 байта

- **Число байт** - количество байт данных в подзапросе.
- **Подзапрос** - группа для чтения, содержащая 7 байт данных для запроса:
  - **Тип ссылки** - тип ссылки на данные, спецификацией определен код «0x06».
  - **Номер файла** - номер файла расширенной памяти.
  - **Номер записи** - адрес регистра внутри файла расширенной памяти.
  - **Длина записи** - количество регистров для чтения.

Входящий кадр функции «20 (0x14)» имеет структуру:

Адрес	Код функции	Число байт	Ответ			Контрольная сумма
			Число байт	Тип ссылки	Данные	
1 байт	1 байт	1 байт	1 байт	1 байт	N байт	2 байта

- **Число байт** - общее количество байт полученных данных.
- **Ответ** - группа полученных данных:
  - **Число байт** - количество байт данных ответа.
  - **Тип ссылки** - тип ссылки на данные, спецификацией определен код «0x06».
  - **Данные** - полученные значения.





## ПРИМЕР

Исходящий кадр функции «20 (0x14) Read File Record»:

Тип ссылки	Номер файла	Номер записи	Длина записи
6	1	0	2

Исходящий кадр. функция 20 (Read File Record)

	0	1	2	3	4	5	6	7	8	9
0000	01	14	07	06	00	01	00	00	00	02
0001	45	25								

Байт	Данные	Описание
0	01	Адрес станции - 1
1	14	Код функции - «20 (0x14)»
2	07	Число байт подзапроса - 7
3	06	Тип ссылки - 6
4, 5	00 01	Номер файла - 1
6, 7	00 00	Номер записи - 0
8, 9	00 02	Длина записи - 2
10, 11	45 25	Контрольная сумма

Входящий кадр функции «20 (0x14) Read File Record»:

Номер записи	Тип ссылки	Hex	Знаковое	Беззнаковое
0	6	A	10	10
1	6	14	20	20

Входящий кадр. функция 20 (Read File Record)

	0	1	2	3	4	5	6	7	8	9
0000	01	14	06	05	06	00	0A	00	14	C9
0001	C7									

Байт	Данные	Описание
0	01	Адрес станции - 1
1	14	Код функции - «20 (0x14)»
2	06	Число байт - 6
3	05	Число байт ответа - 5
4	06	Тип ссылки - 6



Байт	Данные	Описание
5, 6	00 0A	Значение записи 0 - «10»
7, 8	00 14	Значение записи 1 - «20»
9, 10	51 88	Контрольная сумма

### 3.3.1.2. Кадры функций записи

Исходящий кадр для функций «05 (0x05)» и «06 (0x06)» имеет структуру:

Адрес	Код функции	Начальный адрес	Значение	Контрольная сумма
1 байт	1 байт	2 байта	2 байта	2 байта

- **Адрес элемента** - адрес элемента сегмента памяти подчиненной станции, значение которого требуется изменить.
- **Значение** - устанавливаемое значение.

Если функция выполнена успешно, то входящий кадр содержит копию исходящего кадра.



## ПРИМЕР

Исходящий кадр функции «05 (0x05) Write Single Coil»:

Адрес	Hex	Знач...
1	FF00	ON

Исходящий кадр. функция 5 (Write Single Coil)

	0	1	2	3	4	5	6	7	8	9
0000	01	05	00	01	FF	00	DD	FA		

Входящий кадр функции «05 (0x05) Write Single Coil»:

Адрес	Hex	Знач...
1	FF00	ON

Входящий кадр. функция 5 (Write Single Coil)

	0	1	2	3	4	5	6	7	8	9
0000	01	05	00	01	FF	00	DD	FA		

Байт	Данные	Описание
0	01	Адрес станции - 1
1	05	Код функции - «05 (0x05)»
2, 3	00 01	Адрес элемента - 1
4, 5	FF 00	Значение - «ON» (00 00 соответствует значению «OFF»)
6, 7	DD FA	Контрольная сумма

Исходящий кадр функции «16 (0x10)» имеет структуру:

Адрес	Код функции	Начальный адрес	Количество	Число байт	Значения	Контрольная сумма
1 байт	1 байт	2 байта	2 байта	1 байт	N байт	2 байта

- **Начальный адрес** - адрес первого элемента сегмента памяти подчиненной станции, значение которого требуется изменить.
- **Количество** - количество изменяемых элементов.
- **Число байт** - количество передаваемых байт устанавливаемых значений.
- **Значения** - устанавливаемые значения.

Если функция выполнена успешно, то входящий кадр функции «16 (0x10)» имеет структуру:

Адрес	Код функции	Начальный адрес	Количество	Контрольная сумма
1 байт	1 байт	2 байта	N байт	2 байта

- **Начальный адрес** - адрес первого измененного элемента.
- **Количество** - количество измененных элементов.



ПРИМЕР

Исходящий кадр функции «16 (0x10) Write Multiple Registers»:

Адрес	Hex	Знаковое	Беззнаковое
5	2	2	2
6	2334	9012	9012
7	934	2356	2356
8	285C	10332	10332

Исходящий кадр. функция 16 (Write Multiple registers)

	0	1	2	3	4	5	6	7	8	9
0000	01	10	00	05	00	04	08	00	02	23
0001	34	09	34	28	5C	73	4A			

Байт	Данные	Описание
0	01	Адрес станции - 1
1	10	Код функции - «16 (0x10)»
2, 3	00 05	Начальный адрес - 5
4, 5	00 04	Количество - 4
6	08	Число байт - 8
7, 8	00 02	Значение 5 - «2»
9, 10	23 34	Значение 6 - «9012»
11, 12	09 34	Значение 7 - «2356»
12, 14	28 5C	Значение 8 - «10332»
15, 16	73 4A	Контрольная сумма

Входящий кадр функции «16 (0x10) Write Multiple Registers»:

Начальный адрес	Количество
5	4

Входящий кадр. функция 16 (Write Multiple registers)

	0	1	2	3	4	5	6	7	8	9
0000	01	10	00	05	00	04	D1	CB		

Байт	Данные	Описание
0	01	Адрес станции - 1
1	10	Код функции - «16 (0x10)»

Байт	Данные	Описание
2, 3	00 05	Начальный адрес - 5
4, 5	00 04	Количество - 4
6, 7	D1 CB	Контрольная сумма

### 3.3.1.3. Кадры ошибок

Если подчиненная станция не может выполнить функцию, входящий кадр имеет структуру:

Адрес	Код функции + 0x80	Код ошибки	Контрольная сумма
1 байт	1 байт	1 байт	2 байта

В таблице приведены поддерживаемые модулем Modbus RTU Master коды ошибок.

Код	Название	Описание
01	ILLEGAL FUNCTION (Недопустимая функция)	Принятый код функции не может быть обработан подчиненной станцией
02	ILLEGAL DATA ADDRESS (Недопустимый адрес данных)	Адрес регистра подчиненной станции, указанный в запросе, недоступен
03	ILLEGAL DATA VALUE (Недопустимое значение данных)	Значение, указанное в поле данных запроса, является недопустимым
04	SERVER DEVICE FAILURE (Сбой подчиненной станции)	Подчиненная станция не может обработать принятый запрос
06	SERVER DEVICE BUSY (Подчиненная станция занята)	Подчиненная станция занята обработкой команды. Опросчик должен повторить запрос, когда подчиненная станция освободится



ПРИМЕР

Исходящий кадр функции «04 (0x04) Read Input Registers»:

Начальный адрес	Количество
1	2

Исходящий кадр. функция 4 (Read Input Registers)

	0	1	2	3	4	5	6	7	8	9
0000	01	04	00	01	00	02	20	0B		

Байт	Данные	Описание
0	01	Адрес станции - 1
1	04	Код функции - «04 (0x04)»
2, 3	00 01	Начальный адрес - 1
4, 5	00 02	Количество - 2
6, 7	FD D7	Контрольная сумма

Входящий кадр ошибки для функции «04 (0x04) Read Input Registers»:

Код ошибки

2 (Illegal Data Address)

< >

Входящий кадр. Ошибка. Код 2. функция 4 (Read Input Registers)

	0	1	2	3	4	5	6	7	8	9
0000	01	84	02	C2	C1					

Байт	Данные	Описание
0	01	Адрес станции - 1
1	84	Код функции+0x80 - 0x84
2	02	Код ошибки - 2
3, 4	C2 C1	Контрольная сумма

## Список терминов и сокращений

<b>CRC (Cyclical Redundancy Check)</b>	Циклическая проверка четности с избыточностью. Тест, используемый для того, чтобы убедиться в отсутствии ошибок при передаче данных.
<b>LRC (Longitudinal Redundancy Check)</b>	Линейная проверка четности с избыточностью. Тест, используемый для того, чтобы убедиться в отсутствии ошибок при передаче данных.
<b>SCADA (Supervisory Control And Data Acquisition)</b>	Система, обеспечивающая диспетчерское управление и сбор данных.
<b>Активная пара серверов</b>	Состояние резервной пары серверов, при котором может выполняться отправка управляющих и регулирующих воздействий.
<b>Активный запрос</b>	Состояние запроса, определяемое с момента отправки до момента ответа. Сообщение об ошибке принимается за ответ.
<b>АСУ ТП</b>	Автоматизированная система управления технологическим процессом.
<b>Дерево сигналов</b>	Структура технологических данных, с которой работают компоненты АСУ ТП.
<b>Качество сигнала</b>	Свойство сигнала, характеризующее его достоверность.
<b>КП</b>	Контролируемый пункт.
<b>Кратковременная потеря связи</b>	Состояние модуля, когда связь фактически разорвана, но время таймаута потери связи не закончилось.
<b>Метка времени</b>	Время изменения значения или качества сигнала.
<b>Модуль</b>	Программный компонент, работающий в составе сервера, обеспечивающий некоторую логически законченную функциональность. Основной функцией модулей сервера является передача данных между компонентами АСУ ТП на уровне SCADA-системы.
<b>Подчиненная станция</b>	Станция, с которой работает модуль. В роли подчиненной станции может быть либо программируемый логический контролер (ПЛК), либо шлюз, объединяющий несколько ПЛК.
<b>Поллинг</b>	Циклический опрос одного устройства.
<b>Сигнал</b>	Объект, являющийся носителем информации при обмене данными между компонентами АСУ ТП. Сигнал имеет определенный тип и обладает набором свойств. Основное назначение сигналов хранить значения реальных физических величин и их свойства: достоверность, параметры доступа и др.
<b>Таблица поллинга</b>	Набор запросов, циклически отправляемых для чтения данных.
<b>Телеизмерение</b>	Используется для получения количественной оценки характеристик контролируемого процесса, например, температуры, напряжения, тока, давления и пр.

---

<b>Телерегулирование</b>	Обеспечивает дистанционное задание уровня воздействия на объект управления. Управление начинается с задания оператором величины воздействия, а затем выдачей команды с ПК.
<b>Телесигнализация</b>	Используется для дистанционного контроля дискретных изменений состояния объекта, например, включен/выключен, движется/стоит, норма/авария и т.п.
<b>Телесчет</b>	Используется для получения количественной оценки характеристик контролируемого процесса.
<b>Телеуправление</b>	Обеспечивает дистанционное управление объектом контроля. Управление начинается с выдачи оператором (диспетчером) команды телеуправления с ПК или пульта управления.
<b>Транзакция</b>	Запрос, отправленный подчиненной станции.